Longer. Faster. Forever.

Feature Selection

Rais Seminar

Vangjush Komini

KTH

April 1, 2020

racefox

Machine Learning in Nutshell

What is machine?

racefox

"Machine learning research is part of research on artificial intelligence, seeking to provide knowledge to computers through data, observations and interacting with the world. That acquired knowledge allows computers to correctly generalize to new settings."
Yoshua Bengio

What is machine?

racefox

"Machine learning research is part of research on artificial intelligence,seeking to provide knowledge to computers through data,observations and interacting with the world.That acquired knowledge allows computers to correctly generalize to new settings."
Yoshua Bengio

What is machine?

“Machine learning research is part of research on artificial intelligence,seeking to provide knowledge to computers through data,observations and interacting with the world.That acquired knowledge allows computers to correctly generalize to new settings.”
Yoshua Bengio

What is machine?

racefox

"Machine learning research is part of research on artificial intelligence,seeking to provide knowledge to computers through data,observations and interacting with the world.That acquired knowledge allows computers to correctly generalize to new settings."

Yoshua Bengio

What is machine?

racefox

"Machine learning research is part of research on artificial intelligence,seeking to provide knowledge to computers through data,observations and interacting with the world.That acquired knowledge allows computers to correctly generalize to new settings."
Yoshua Bengio

Why we need ML?

racefox

- Significant amounts of data are available or can be generated (either beforehand or dynamically)
- Other (analytical) solutions are too slow or infeasible
- Human expertise is absent or unexplainable
- Solutions change over time or need to be adapted
- ....

From DD3359-RL course KTH

Why we need ML?

racefox

- Significant amounts of data are available or can be generated (either beforehand or dynamically)
- Other (analytical) solutions are too slow or infeasible
- Human expertise is absent or unexplainable
- Solutions change over time or need to be adapted
- ....

From DD3359-RL course KTH

Why we need ML?

racefox

- Significant amounts of data are available or can be generated (either beforehand or dynamically)
- Other (analytical) solutions are too slow or infeasible
- Human expertise is absent or unexplainable
- Solutions change over time or need to be adapted
- ....

From DD3359-RL course KTH

Why we need ML?

- Significant amounts of data are available or can be generated (either beforehand or dynamically)
- Other (analytical) solutions are too slow or infeasible
- Human expertise is absent or unexplainable
- Solutions change over time or need to be adapted
- ....

From DD3359-RL course KTH

Why we need ML?

racefox

- Significant amounts of data are available or can be generated (either beforehand or dynamically)
- Other (analytical) solutions are too slow or infeasible
- Human expertise is absent or unexplainable
- Solutions change over time or need to be adapted
- ....

From DD3359-RL course KTH

What is DL?

racefox

Deep learning allows computational models, that are composed of multiple processing layers to learn representations of data, with multiple levels of abstraction.

Y.LeCun

What is DL?

Deep learning allows computational models,that are composed of multiple processing layers to learn representations of data, with multiple levels of abstraction.

Y.LeCun

What is DL?

racefox

Deep learning allows computational models, that are composed of multiple processing layers to learn representations of data, with multiple levels of abstraction.

Y.LeCun

What is DL?

racefox

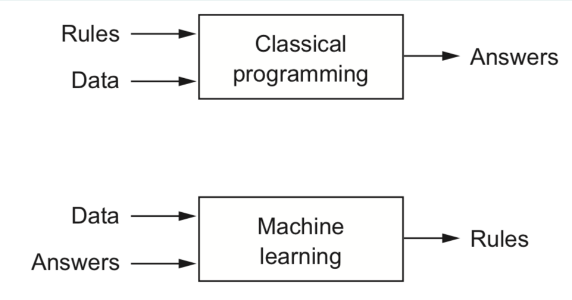
Deep learning allows computational models,that are composed of multiple processing layers to learn representations of data, with multiple levels of abstraction.

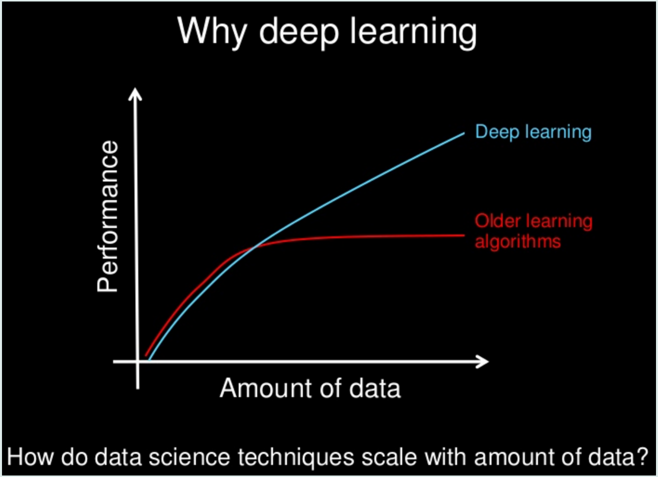
Y.LeCun

What is DL?

racefox

Deep learning allows computational models,that are composed of multiple processing layers to learn representations of data, with multiple levels of abstraction.
Y.LeCun


Rules
Data
Classical programming
Answers
Data
Answers
Machine learning
Rules


Why deep learning
Performance
Deep learning
Older learning algorithms
Amount of data
How do data science techniques scale with amount of data?

Generalization vs Structuring

racefox

- Supervised learning tries to generalize over an massive amount of **structured** data.
- Unsupervised learning tries to learn the **structure** of a massive amount of data.
  - Clustering tries to bring together items with high similarity of **invarian** features.
  - Density estimation tries to model a probability distribution of the items influenced by the **invariant** features (Central Limit Theorem to be considered).
  - Dimensionality reduction find the a latent space where the **invariant** features prevail.
- Semi-/Weakly- supervised learning tries to learn the **scarcely** labeled data.
- Individual data is assumed to be **composed** of **core content** which is **invariant** from the **acquisition conditions** and the **non-core** content **dependent acquisition conditions**.

Generalization vs Structuring

racefox

- Supervised learning tries to generalize over an massive amount of **structured** data.
- Unsupervised learning tries to learn the **structure** of a massive amount of data.
  - Clustering tries to bring together items with high similarity of **invarian** features.
  - Density estimation tries to model a probability distribution of the items influenced by the **invariant** features (Central Limit Theorem to be considered).
  - Dimensionality reduction find the a latent space where the **invariant** features prevail.
- Semi-/Weakly- supervised learning tries to learn the **scarcely** labeled data.
- Individual data is assumed to be **composed** of **core content** which is **invariant** from the **acquisition conditions** and the **non-core** content **dependent acquisition conditions**.

Generalization vs Structuring

racefox

- Supervised learning tries to generalize over an massive amount of **structured** data.
- Unsupervised learning tries to learn the **structure** of a massive amount of data.
  - Clustering tries to bring together items with high similarity of **invarian** features.
  - Density estimation tries to model a probability distribution of the items influenced by the **invariant** features (Central Limit Theorem to be considered).
  - Dimensionality reduction find the a latent space where the **invariant** features prevail.
- Semi-/Weakly- supervised learning tries to learn the **scarcely** labeled data.
- Individual data is assumed to be **composed** of **core content** which is **invariant** from the **acquisition conditions** and the **non-core** content **dependent acquisition conditions**.

Generalization vs Structuring

racefox

- Supervised learning tries to generalize over an massive amount of **structured** data.
- Unsupervised learning tries to learn the **structure** of a massive amount of data.
  - Clustering tries to bring together items with high similarity of **invarian** features.
  - Density estimation tries to model a probability distribution of the items influenced by the **invariant** features (Central Limit Theorem to be considered).
  - Dimensionality reduction find the a latent space where the **invariant** features prevail.
- Semi-/Weakly- supervised learning tries to learn the **scarcely** labeled data.
- Individual data is assumed to be **composed** of **core content** which is **invariant** from the **acquisition conditions** and the **non-core** content **dependent acquisition conditions**.

Generalization vs Structuring

racefox

- Supervised learning tries to generalize over an massive amount of **structured** data.
- Unsupervised learning tries to learn the **structure** of a massive amount of data.
  - Clustering tries to bring together items with high similarity of **invarian** features.
  - Density estimation tries to model a probability distribution of the items influenced by the **invariant** features (Central Limit Theorem to be considered).
  - Dimensionality reduction find the a latent space where the **invariant** features prevail.
- Semi-/Weakly- supervised learning tries to learn the **scarcely** labeled data.
- Individual data is assumed to be **composed** of **core content** which is **invariant** from the **acquisition conditions** and the **non-core** content **dependent acquisition conditions**.

Generalization vs Structuring

racefox

- Supervised learning tries to generalize over an massive amount of **structured** data.
- Unsupervised learning tries to learn the **structure** of a massive amount of data.
  - Clustering tries to bring together items with high similarity of **invarian** features.
  - Density estimation tries to model a probability distribution of the items influenced by the **invariant** features (Central Limit Theorem to be considered).
  - Dimensionality reduction find the a latent space where the **invariant** features prevail.
- Semi-/Weakly- supervised learning tries to learn the **scarcely** labeled data.
- Individual data is assumed to be **composed** of **core content** which is **invariant** from the **acquisition conditions** and the **non-core** content **dependent acquisition conditions**.

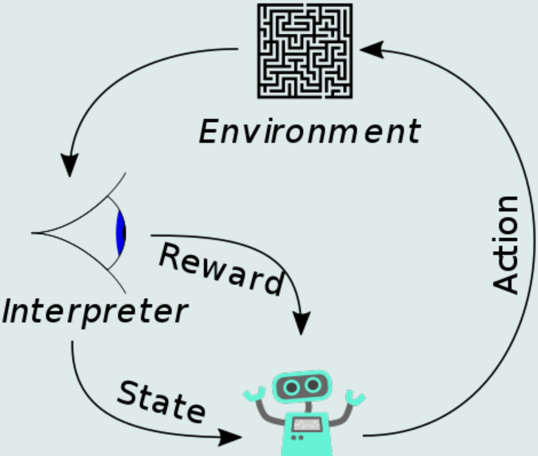
Generalization vs Structuring

racefox

- Supervised learning tries to generalize over an massive amount of **structured** data.
- Unsupervised learning tries to learn the **structure** of a massive amount of data.
  - Clustering tries to bring together items with high similarity of **invarian** features.
  - Density estimation tries to model a probability distribution of the items influenced by the **invariant** features (Central Limit Theorem to be considered).
  - Dimensionality reduction find the a latent space where the **invariant** features prevail.
- Semi-/Weakly- supervised learning tries to learn the **scarcely** labeled data.
- Individual data is assumed to be **composed** of **core content** which is **invariant** from the **acquisition conditions** and the **non-core** content **dependent acquisition conditions**.

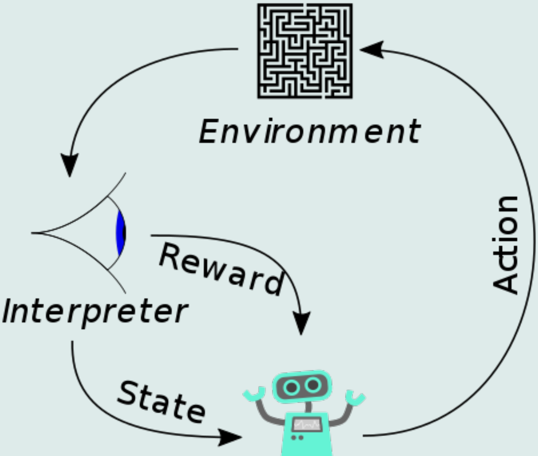
Other type

- Reinforcement learning: Trying to generalize of a series of sequential observations from an enviornment by learning a policy generates a given action from the state, such as the cummulative rewards (sparse in time) are maximized.
- More biologically plausiblea approach.



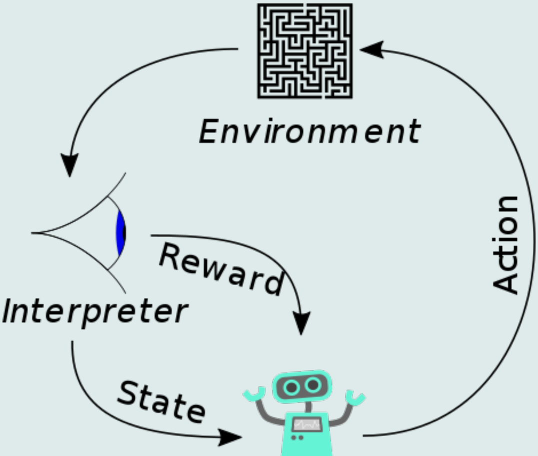
Other type

- Reinforcement learning: Trying to generalize of a series of sequential observations from an enviornment by learning a policy generates a given action from the state, such as the cummulative rewards (sparse in time) are maximized.
- More biologically plausiblea approach.



Other type

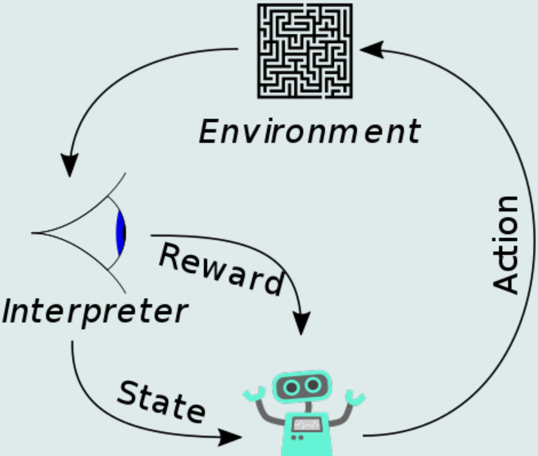
- Reinforcement learning: Trying to generalize of a series of sequential observations from an enviornment by learning a policy generates a given action from the state, such as the cummulative rewards (sparse in time) are maximized.
- More biologically plausiblea approach.



Other type

racefox

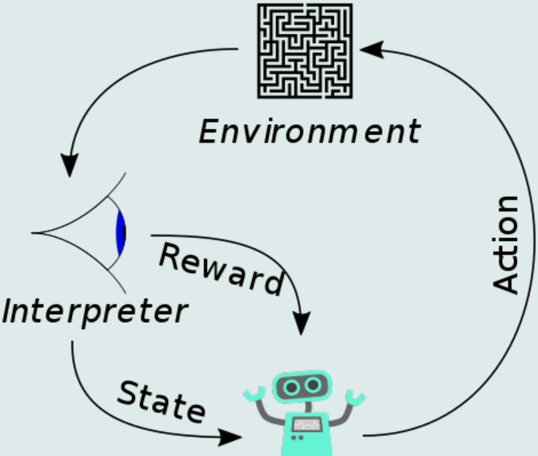
- Reinforcement learning: Trying to generalize of a series of sequential observations from an enviornment by learning a policy generates a given action from the state, such as the cummulative rewards (sparse in time) are maximized.
- More biologically plausiblea approach.



Other type

racefox

- Reinforcement learning: Trying to generalize of a series of sequential observations from an enviornment by learning a policy generates a given action from the state, such as the cummulative rewards (sparse in time) are maximized.
- More biologically plausiblea approach.



Machine Learning in practice.

Treating the concept as a mathematical computable entity, and sampling a lot of data from the this entity, and use these empirical data as a proxy.

Machine Learning in practice.

racefox

Treating the concept as a mathematical computable entity,and sampling a lot of data from the this entity,and use these empirical data as a proxy.

Machine Learning in practice.

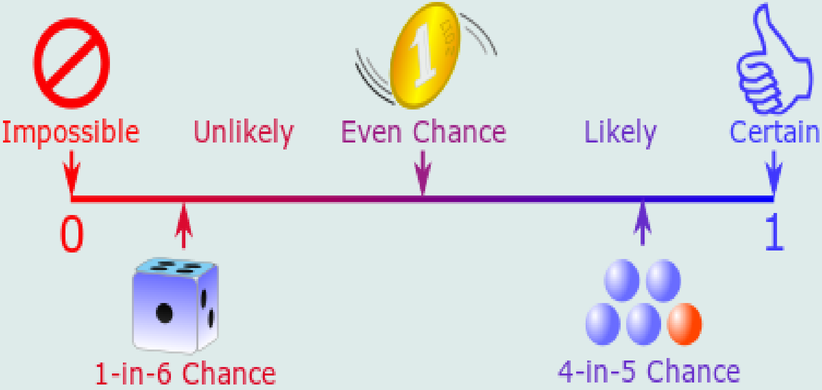
racefox

Treating the concept as a mathematical computable entity,and sampling a lot of data from the this entity,and use these empirical data as a proxy.

Key Mathematical Ingredients

racefox

- Probability: the calculus of **uncertainity** computation.
- Calculus: the science of **continuity** that is at continous **change**.
- Algebra: the science of **multidimensional** hyper-space.
- Graphs: the science **ontological** entities.
- ....



Key Mathematical Ingredients

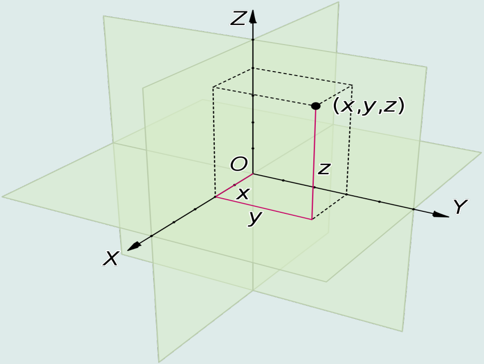
racefox

- Probability: the calculus of **uncertainity** computation.
- Calculus: the science of **continuity** that is at continous **change**.
- Algebra: the science of **multidimensional** hyper-space.
- Graphs: the science **ontological** entities.
- ....

Key Mathematical Ingredients

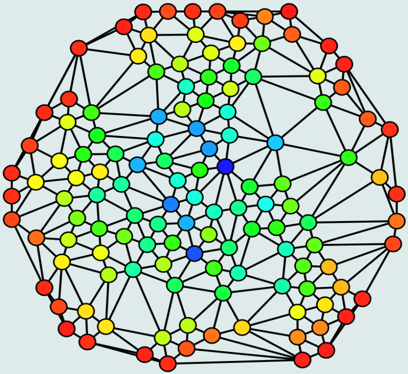
racefox

- Probability: the calculus of **uncertainity** computation.
- Calculus: the science of **continuity** that is at continous **change**.
- Algebra: the science of **multidimensional** hyper-space.
- Graphs: the science **ontological** entities.
- ....



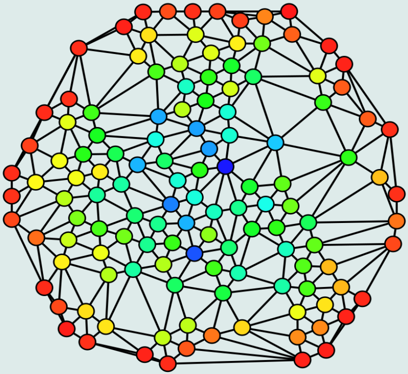
Key Mathematical Ingredients

- Probability: the calculus of **uncertainity** computation.
- Calculus: the science of **continuity** that is at continous **change**.
- Algebra: the science of **multidimensional** hyper-space.
- Graphs: the science **ontological** entities.
- ....



Key Mathematical Ingredients

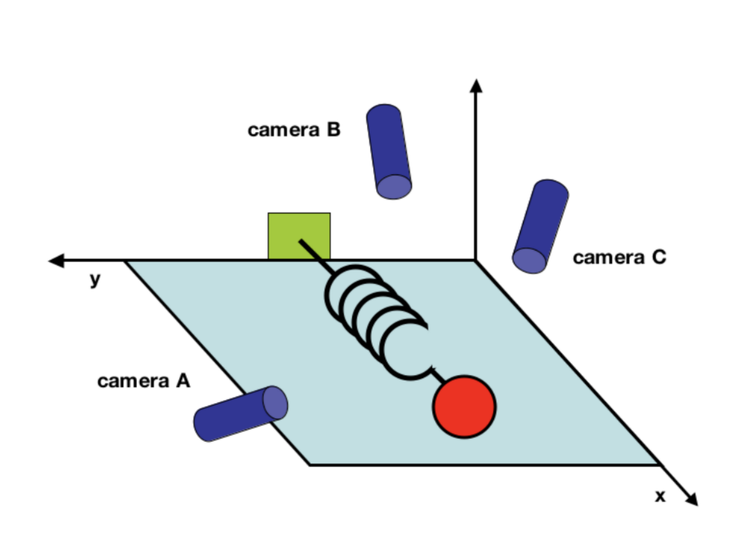
- Probability: the calculus of **uncertainity** computation.
- Calculus: the science of **continuity** that is at continous **change**.
- Algebra: the science of **multidimensional** hyper-space.
- Graphs: the science **ontological** entities.
- ....



Feature selection

Which camera is better one

racefox

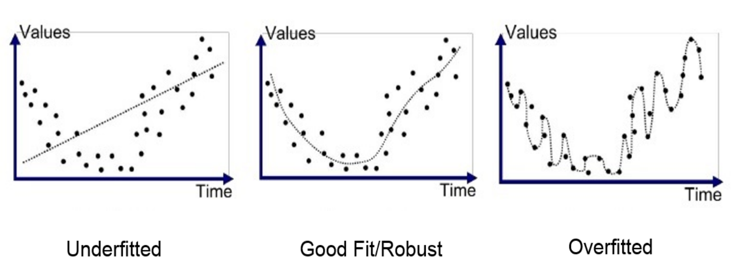


Shlens et al 2013

Why we need feature selection

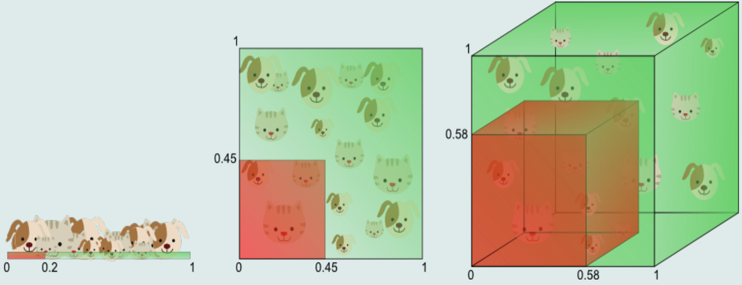
racefox

- reducing overfitting
- overcoming the curse of dimensionality
- shorter training time
- improve the interpretability of the methods
- Other?



Why we need feature selection

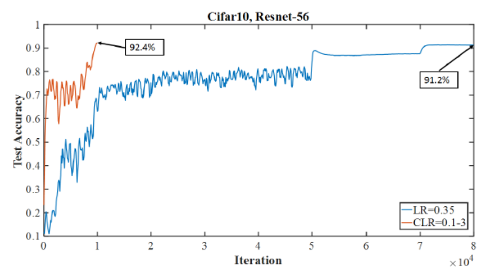
- reducing overfitting
- overcoming the curse of dimensionality
- shorter training time
- improve the interpretability of the methods
- Other?



Why we need feature selection

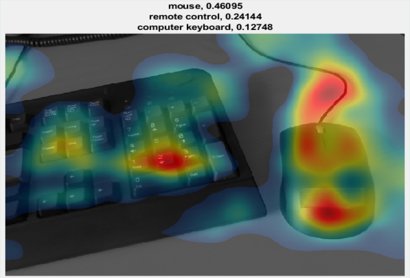
racefox

- reducing overfitting
- overcoming the curse of dimensionality
- shorter training time
- improve the interpretability of the methods
- Other?



Why we need feature selection

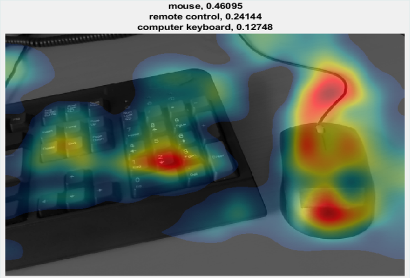
- reducing overfitting
- overcoming the curse of dimensionality
- shorter training time
- improve the interpretability of the methods
- Other?



Why we need feature selection

racefox

- reducing overfitting
- overcoming the curse of dimensionality
- shorter training time
- improve the interpretability of the methods
- Other?



Feature selections, Chen et al

racefox

- Let $X \subset R^d$ be the domain of covariates.
- Let $Y \subset 0, 1$ be the domain of responses (labels).
- Given n i.i.d data pairs $\{(x_i, y_i), = 1, 2, .., d\}$, with unknown distribution P(X,Y)
- Select a subset of X that best predict Y.

Feature selections, Chen et al

- Let $X \subset R^d$ be the domain of covariates.
- Let $Y \subset 0, 1$ be the domain of responses (labels).
- Given n i.i.d data pairs $\{(x_i, y_i), = 1, 2, .., d\}$, with unknown distribution P(X,Y)
- Select a subset of X that best predict Y.

Feature selections, Chen et al

racefox

- Let $X \subset R^d$ be the domain of covariates.
- Let $Y \subset 0, 1$ be the domain of responses (labels).
- Given *n* i.i.d data pairs $\{(x_i, y_i), = 1, 2, .., d\}$, with unknown distribution P(X,Y)
- Select a subset of X that best predict Y.

Feature selections, Chen et al

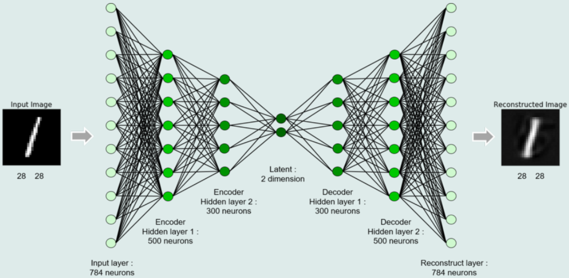
racefox

- Let $X \subset R^d$ be the domain of covariates.
- Let $Y \subset 0, 1$ be the domain of responses (labels).
- Given n i.i.d data pairs $\{(x_i, y_i), = 1, 2, .., d\}$, with unknown distribution P(X,Y)
- Select a subset of X that best predict Y.

Concrete autoencoder,Balin et al 2019

Concrete autoencoder

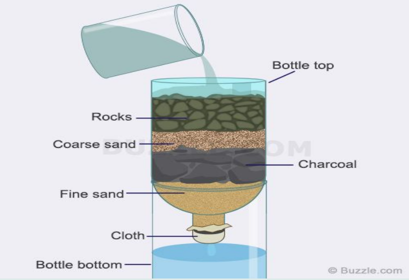
- Utilization of autoencoders, for distillation of predictive features.
- Latent space could be any type of mathematical entity.
- Reparameterization enables back-propagation in random variables.
- Concrete autoncoder is still an autoencoder.



Concrete autoencoder

racefox

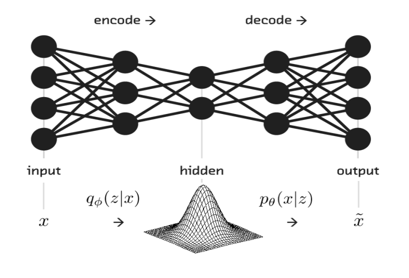
- Utilization of autoencoders, for distillation of predictive features.
- Latent space could be any type of mathematical entity.
- Reparameterization enables back-propagation in random variables.
- Concrete autoncoder is still an autoencoder.



Concrete autoencoder

racefox

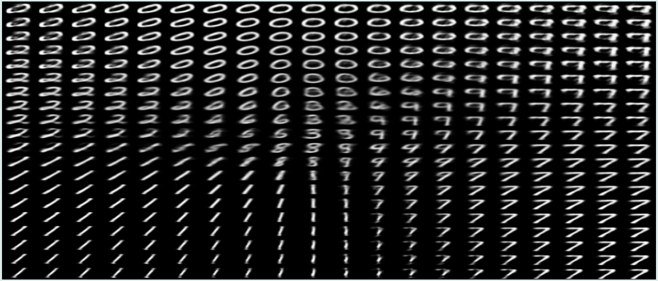
- Utilization of autoencoders, for distillation of predictive features.
- Latent space could be any type of mathematical entity.
- Reparameterization enables back-propagation in random variables.
- Concrete autoncoder is still an autoencoder.



Concrete autoencoder

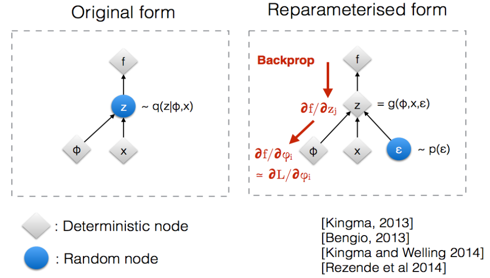
racefox

- Utilization of autoencoders, for distillation of predictive features.
- Latent space could be any type of mathematical entity.
- Reparameterization enables back-propagation in random variables.
- Concrete autoncoder is still an autoencoder.



Concrete autoencoder

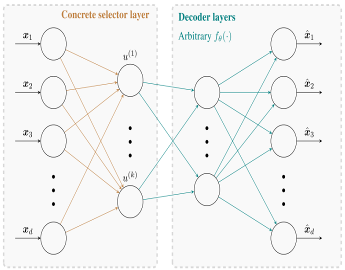
- Utilization of autoencoders, for distillation of predictive features.
- Latent space could be any type of mathematical entity.
- Reparameterization enables back-propagation in random variables.
- Concrete autoncoder is still an autoencoder.



Concrete autoencoder

racefox

- Utilization of autoencoders, for distillation of predictive features.
- Latent space could be any type of mathematical entity.
- Reparameterization enables back-propagation in random variables.
- Concrete autoncoder is still an autoencoder.



Algorithm 1. Training a Concrete Autoencoder

Input: training dataset $X \in \mathbb{R}^{n \times d}$, number of features to select k, decoder network $f_\theta(\cdot)$, number of epochs B, learning rate λ, initial temp. T_0, final temp. T_B.

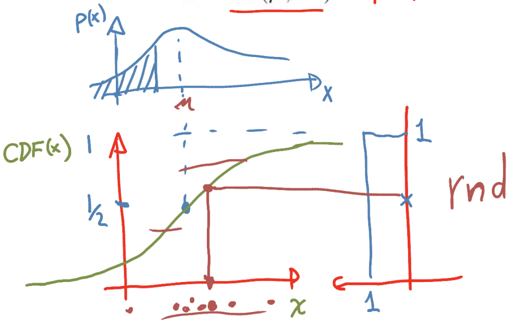
Initialize the parameters of the concrete autoencoder.
for $b \in \{1 \dots B\}$ **do**
  Adjust temp. of Concrete dist. $T = T_0(T_B/T_0)^{b/B}$
  for $i \in \{1 \dots k\}$ **do**
    Sample $m^{(i)} \sim \text{Concrete}(\alpha^{(i)}, T)$
    Let $u^{(i)} = X \cdot m^{(i)}$
  end for
  Define U to be the matrix $\in \mathbb{R}^{n \times k}$ that results from horizontally concatenating the $u^{(1)} \dots u^{(k)}$.
  Let the loss L be defined as $\|f_\theta(U) - X\|_F$
  Compute gradient of the loss w.r.t. θ using backpropagation and w.r.t each $\alpha^{(i)}$ using reparametrization trick.
  Update $\theta \leftarrow \theta - \lambda \nabla_\theta L$, and $\alpha^{(i)} \leftarrow \alpha^{(i)} - \lambda \nabla_{\alpha^{(i)}} L$
end for

Return: decoder $f_\theta(\cdot)$ and Concrete params. $\alpha^{(i)}$

Sampling from a Gaussian distribution

Simulate / imagine / hallucinate / draw

$$x \sim \underline{\mathcal{N}(\mu, \sigma^2)} \equiv p(x)$$



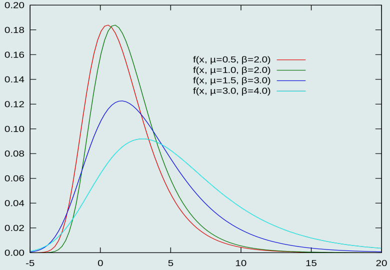
Sample $x^{(i)} \sim p(x)$

Slides taken from Nando.D.F course Machine Learning: 2014-2015

Concrete distribution,Maddison et al 2017

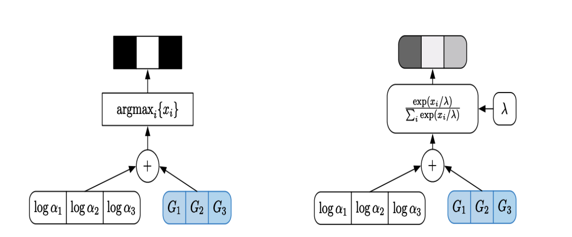
racefox

- Sample and array g_j from a Gumbel distribution $F(x;\mu,\beta) = e^{-e^{-\frac{x-\mu}{\beta}}}$
- Relaxation of the discrete variables $m_j = \frac{e^{(log(\alpha_j)+g_j)/\lambda}}{\sum_{k=1}^{d} e^{(log(\alpha_k)+g_k)\lambda}}$
- One-hot encoding distribution $m_j = 1, with, probability = \frac{\alpha_j}{\sum_{k=1}^{d} \alpha_k}$
- Temperature modulation $\lambda(epoch) = \lambda_{initial}\left\{\frac{\lambda_{final}}{\lambda_{initial}}\right\}^{\left\{\frac{epoch}{TotalNrEpochs}\right\}}$



Concrete distribution,Maddison et al 2017

- Sample and array g_j from a Gumbel distribution $F(x;\mu,\beta) = e^{-e^{-\frac{x-\mu}{\beta}}}$
- Relaxation of the discrete variables $m_j = \frac{e^{(log(\alpha_j)+g_j)/\lambda}}{\sum_{k=1}^{d} e^{(log(\alpha_k)+g_k)\lambda}}$
- One-hot encoding distribution $m_j = 1, with, probability = \frac{\alpha_j}{\sum_{k=1}^{d} \alpha_k}$
- Temperature modulation $\lambda(epoch) = \lambda_{initial} \left\{ \frac{\lambda_{final}}{\lambda_{initial}} \right\}^{\left\{ \frac{epoch}{TotalNrEpochs} \right\}}$



(a) Discrete(α) (b) Concrete(α, λ)

Concrete distribution,Maddison et al 2017

- Sample and array g_j from a Gumbel distribution $F(x;\mu,\beta) = e^{-e^{-\frac{x-\mu}{\beta}}}$
- Relaxation of the discrete variables $m_j = \frac{e^{(log(\alpha_j)+g_j)/\lambda}}{\sum_{k=1}^{d} e^{(log(\alpha_k)+g_k)\lambda}}$
- One-hot encoding distribution $m_j = 1, with, probability = \frac{\alpha_j}{\sum_{k=1}^{d} \alpha_k}$
- Temperature modulation $\lambda(epoch) = \lambda_{initial}\left\{\frac{\lambda_{final}}{\lambda_{initial}}\right\}^{\left\{\frac{epoch}{TotalNrEpochs}\right\}}$

OneHot Encoding

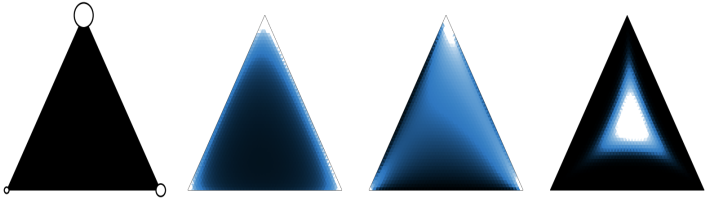
workclass
State-gov
Self-emp-not-inc
Private
Private
Private

State-gov	Self-emp-not-inc	Private
1	0	0
0	1	0
0	0	1
0	0	1
0	0	1

Concrete distribution,Maddison et al 2017

racefox

- Sample and array g_j from a Gumbel distribution $F(x;\mu,\beta) = e^{-e^{-\frac{x-\mu}{\beta}}}$
- Relaxation of the discrete variables $m_j = \frac{e^{(log(\alpha_j)+g_j)/\lambda}}{\sum_{k=1}^{d} e^{(log(\alpha_k)+g_k)\lambda}}$
- One-hot encoding distribution $m_j = 1, with, probability = \frac{\alpha_j}{\sum_{k=1}^{d} \alpha_k}$
- Temperature modulation $\lambda(epoch) = \lambda_{initial}\left\{\frac{\lambda_{final}}{\lambda_{initial}}\right\}^{\left\{\frac{epoch}{TotalNrEpochs}\right\}}$

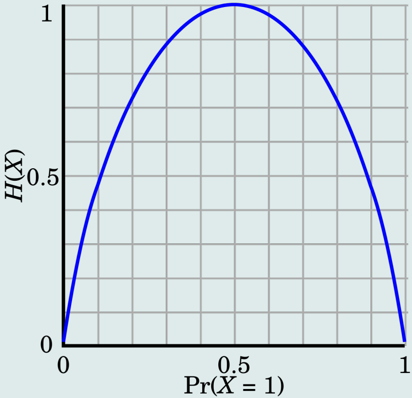


(a) $\lambda = 0$ (b) $\lambda = 1/2$ (c) $\lambda = 1$ (d) $\lambda = 2$

Variational Information Maximization for Feature Selection Gao et al

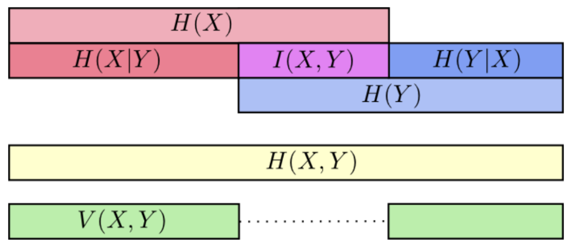
Entropy

racefox



$$Entropy : H(X) = -\sum_i p_X(x_i) * log(p_X(x_i)) \qquad (1)$$

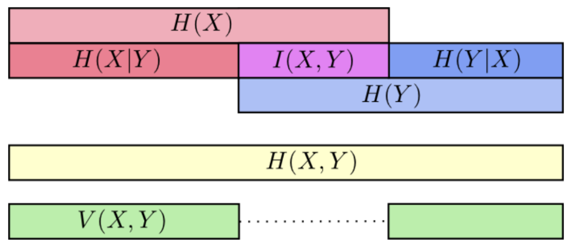
Mutual Information



https://colah.github.io/

- $I(X) = H(X) + H(Y) - H(X,Y)$
- $VI(X,Y) = H(X,Y) - I(X,Y)$
- $D_{KL}(P_X||Q_X) = \int_{-\infty}^{+\infty} p(x) log \left\{ \frac{p(x)}{q(x)} \right\} dx$

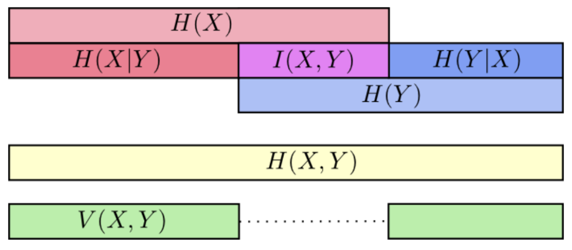
Mutual Information



https://colah.github.io/

- $I(X) = H(X) + H(Y) - H(X,Y)$
- $VI(X,Y) = H(X,Y) - I(X,Y)$
- $D_{KL}(P_X||Q_X) = \int_{-\infty}^{+\infty} p(x) log \left\{ \frac{p(x)}{q(x)} \right\} dx$

Mutual Information

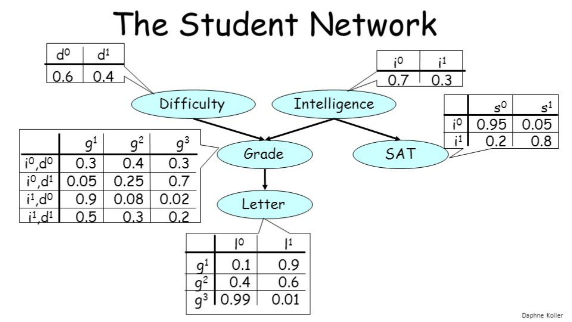


https://colah.github.io/

- $I(X) = H(X) + H(Y) - H(X,Y)$
- $VI(X,Y) = H(X,Y) - I(X,Y)$
- $D_{KL}(P_X||Q_X) = \int_{-\infty}^{+\infty} p(x) log \left\{ \frac{p(x)}{q(x)} \right\} dx$

Probabilistc Graphical Model

racefox



d^0	d^1
0.6	0.4

i^0	i^1
0.7	0.3

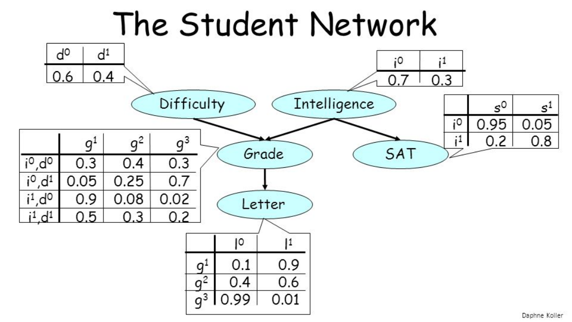
	g^1	g^2	g^3
i^0,d^0	0.3	0.4	0.3
i^0,d^1	0.05	0.25	0.7
i^1,d^0	0.9	0.08	0.02
i^1,d^1	0.5	0.3	0.2

	s^0	s^1
i^0	0.95	0.05
i^1	0.2	0.8

	l^0	l^1
g^1	0.1	0.9
g^2	0.4	0.6
g^3	0.99	0.01

- *Bayes* : $P(\theta|\mathbf{Data}) = \frac{P(\theta,\mathbf{Data})}{P(\mathbf{Data})} = \frac{P(\theta)*P(\mathbf{Data}|\theta)}{P(\mathbf{Data})}$
- $P(X|Y) = P(X) => P(X,Y) = P(X)P(Y)$

Probabilistc Graphical Model



- *Bayes* : $P(\theta|\mathbf{Data}) = \frac{P(\theta,\mathbf{Data})}{P(\mathbf{Data})} = \frac{P(\theta)*P(\mathbf{Data}|\theta)}{P(\mathbf{Data})}$
- $P(X|Y) = P(X) => P(X,Y) = P(X)P(Y)$

Feature selections dependence perspective

- We would like a subset T of size $\textbf{(}$m)
  s.t the remaining S T are conditionally independent given T.
- This dependency is quantified by $Q : 2^d \to [0, \infty)$ such that:
  Q(T)=0 iff $X_{S \ T} \perp Y | X_T$
  $Q(T) \geq Q(S)$ whenever $T \subset S$

$$min_{T:|T|=m} Q(T) \tag{2}$$

Feature selections dependence perspective

racefox

- We would like a subset T of size **(**m)
  s.t the remaining S T are conditionally independent given T.
- This dependency is quantified by $Q : 2^d \rightarrow [0, \infty)$ such that:
  Q(T)=0 iff $X_{S\ T} \perp Y|X_T$
  $Q(T) \geq Q(S)$ whenever $T \subset S$

$$min_{T:|T|=m} Q(T) \tag{2}$$

Feature selections dependence perspective

racefox

- We would like a subset T of size $\mathbf{(}$m)
  s.t the remaining S T are conditionally independent given T.
- This dependency is quantified by $Q : 2^d \to [0, \infty)$ such that:
  Q(T)=0 iff $X_{S \ T} \perp Y | X_T$
  $Q(T) \geq Q(S)$ whenever $T \subset S$

$$min_{T:|T|=m} Q(T) \tag{2}$$

Feature selections dependence perspective

racefox

- We would like a subset T of size (m)
  s.t the remaining S T are conditionally independent given T.
- This dependency is quantified by $Q : 2^d \rightarrow [0, \infty)$ such that:
  Q(T)=0 iff $X_{S\ T} \perp Y | X_T$
  $Q(T) \geq Q(S)$ whenever $T \subset S$

$$min_{T:|T|=m} Q(T) \tag{2}$$

Feature just random variables

racefox

- $T = argmax_T \left\{ I\{(x_1, x_2, ...x_T), y\} \right\}$ NP-hard direct solution
- Forward Feature Selection :$t_{step=t} = argmax_{i \notin S^{t-1}} \left\{ I(x_{S^{t-1} \cup i} : y) \right\}$
- $I(x_{S^{t-1} \cup i} : y) = I(x_{S^{t-1}} : y) + I(x_i : y | x_{S^{t-1}})$
- $I(x_{S^{t-1} \cup i} : y) = I(x_{S^{t-1}} : y) + I(x_i : y) - I(x_i : x_{S^{t-1}}) + I(x_i : x_{S^{t-1}} | y)$
- $= I(x_{S^{t-1}} : y) + I(x_i : y) - (H(x_{S^{t-1}}) - H(x_{S^{t-1}} | x_i)) + (H(x_{S^{t-1}} | y) - H(x_{S^{t-1}} | x_i, y))$
- $t = argmax_{i \notin S^{t-1}} \left\{ I(x_i : y) + H(x_{S^{t-1}} | x_i) - H(x_{S^{t-1}} | x_i, y) \right\}$
- $H(x_{S^{t-1}} | x_i) \approx \sum_{k=1}^{t-1} H(x_k | x_i)$
- $H(x_{S^{t-1}} | x_i, y) \approx \sum_{k=1}^{t-1} H(x_k | x_i, y)$

Not intended to be understood at a single slide. Check reference for further understanding.

Feature just random variables

racefox

- $T = argmax_T \left\{ I\{(x_1, x_2, ...x_T), y\} \right\}$ NP-hard direct solution
- Forward Feature Selection : $t_{step=t} = argmax_{i \notin S^{t-1}} \left\{ I(x_{S^{t-1} \cup i} : y) \right\}$
- $I(x_{S^{t-1} \cup i} : y) = I(x_{S^{t-1}} : y) + I(x_i : y | x_{S^{t-1}})$
- $I(x_{S^{t-1} \cup i} : y) = I(x_{S^{t-1}} : y) + I(x_i : y) - I(x_i : x_{S^{t-1}}) + I(x_i : x_{S^{t-1}} | y)$
- $= I(x_{S^{t-1}} : y) + I(x_i : y) - (H(x_{S^{t-1}}) - H(x_{S^{t-1}} | x_i)) + (H(x_{S^{t-1}} | y) - H(x_{S^{t-1}} | x_i, y))$
- $t = argmax_{i \notin S^{t-1}} \left\{ I(x_i : y) + H(x_{S^{t-1}} | x_i) - H(x_{S^{t-1}} | x_i, y) \right\}$
- $H(x_{S^{t-1}} | x_i) \approx \sum_{k=1}^{t-1} H(x_k | x_i)$
- $H(x_{S^{t-1}} | x_i, y) \approx \sum_{k=1}^{t-1} H(x_k | x_i, y)$

Not intended to be understood at a single slide. Check reference for further understanding.

Feature just random variables

racefox

- $T = argmax_T \left\{ I\{(x_1, x_2, ...x_T), y\} \right\}$ NP-hard direct solution
- Forward Feature Selection : $t_{step=t} = argmax_{i \notin S^{t-1}} \left\{ I(x_{S^{t-1} \cup i} : y) \right\}$
- $I(x_{S^{t-1} \cup i} : y) = I(x_{S^{t-1}} : y) + I(x_i : y | x_{S^{t-1}})$
- $I(x_{S^{t-1} \cup i} : y) = I(x_{S^{t-1}} : y) + I(x_i : y) - I(x_i : x_{S^{t-1}}) + I(x_i : x_{S^{t-1}} | y)$
- $= I(x_{S^{t-1}} : y) + I(x_i : y) - (H(x_{S^{t-1}}) - H(x_{S^{t-1}} | x_i)) + (H(x_{S^{t-1}} | y) - H(x_{S^{t-1}} | x_i, y))$
- $t = argmax_{i \notin S^{t-1}} \left\{ I(x_i : y) + H(x_{S^{t-1}} | x_i) - H(x_{S^{t-1}} | x_i, y) \right\}$
- $H(x_{S^{t-1}} | x_i) \approx \sum_{k=1}^{t-1} H(x_k | x_i)$
- $H(x_{S^{t-1}} | x_i, y) \approx \sum_{k=1}^{t-1} H(x_k | x_i, y)$

Not intended to be understood at a single slide. Check reference for further understanding.

Feature just random variables

- $T = argmax_T \left\{ I\{(x_1, x_2, ...x_T), y\} \right\}$ NP-hard direct solution
- Forward Feature Selection $: t_{step=t} = argmax_{i \notin S^{t-1}} \left\{ I(x_{S^{t-1} \cup i} : y) \right\}$
- $I(x_{S^{t-1} \cup i} : y) = I(x_{S^{t-1}} : y) + I(x_i : y | x_{S^{t-1}})$
- $I(x_{S^{t-1} \cup i} : y) = I(x_{S^{t-1}} : y) + I(x_i : y) - I(x_i : x_{S^{t-1}}) + I(x_i : x_{S^{t-1}} | y)$
- $= I(x_{S^{t-1}} : y) + I(x_i : y) - (H(x_{S^{t-1}}) - H(x_{S^{t-1}} | x_i)) + (H(x_{S^{t-1}} | y) - H(x_{S^{t-1}} | x_i, y))$
- $t = argmax_{i \notin S^{t-1}} \left\{ I(x_i : y) + H(x_{S^{t-1}} | x_i) - H(x_{S^{t-1}} | x_i, y) \right\}$
- $H(x_{S^{t-1}} | x_i) \approx \sum_{k=1}^{t-1} H(x_k | x_i)$
- $H(x_{S^{t-1}} | x_i, y) \approx \sum_{k=1}^{t-1} H(x_k | x_i, y)$

Not intended to be understood at a single slide. Check reference for further understanding.

Feature just random variables

racefox

- $T = argmax_T \left\{ I\{(x_1, x_2, ...x_T), y\} \right\}$ NP-hard direct solution
- Forward Feature Selection :$t_{step=t} = argmax_{i \notin S^{t-1}} \left\{ I(x_{S^{t-1} \cup i} : y) \right\}$
- $I(x_{S^{t-1} \cup i} : y) = I(x_{S^{t-1}} : y) + I(x_i : y | x_{S^{t-1}})$
- $I(x_{S^{t-1} \cup i} : y) = I(x_{S^{t-1}} : y) + I(x_i : y) - I(x_i : x_{S^{t-1}}) + I(x_i : x_{S^{t-1}} | y)$
- $= I(x_{S^{t-1}} : y) + I(x_i : y) - (H(x_{S^{t-1}}) - H(x_{S^{t-1}} | x_i)) + (H(x_{S^{t-1}} | y) - H(x_{S^{t-1}} | x_i, y))$
- $t = argmax_{i \notin S^{t-1}} \left\{ I(x_i : y) + H(x_{S^{t-1}} | x_i) - H(x_{S^{t-1}} | x_i, y) \right\}$
- $H(x_{S^{t-1}} | x_i) \approx \sum_{k=1}^{t-1} H(x_k | x_i)$
- $H(x_{S^{t-1}} | x_i, y) \approx \sum_{k=1}^{t-1} H(x_k | x_i, y)$

Not intended to be understood at a single slide. Check reference for further understanding.

Feature just random variables

racefox

- $T = argmax_T \left\{ I\{(x_1, x_2, ... x_T), y\} \right\}$ NP-hard direct solution
- Forward Feature Selection $: t_{step=t} = argmax_{i \notin S^{t-1}} \left\{ I(x_{S^{t-1} \cup i} : y) \right\}$
- $I(x_{S^{t-1} \cup i} : y) = I(x_{S^{t-1}} : y) + I(x_i : y | x_{S^{t-1}})$
- $I(x_{S^{t-1} \cup i} : y) = I(x_{S^{t-1}} : y) + I(x_i : y) - I(x_i : x_{S^{t-1}}) + I(x_i : x_{S^{t-1}} | y)$
- $= I(x_{S^{t-1}} : y) + I(x_i : y) - (H(x_{S^{t-1}}) - H(x_{S^{t-1}} | x_i)) + (H(x_{S^{t-1}} | y) - H(x_{S^{t-1}} | x_i, y))$
- $t = argmax_{i \notin S^{t-1}} \left\{ I(x_i : y) + H(x_{S^{t-1}} | x_i) - H(x_{S^{t-1}} | x_i, y) \right\}$
- $H(x_{S^{t-1}} | x_i) \approx \sum_{k=1}^{t-1} H(x_k | x_i)$
- $H(x_{S^{t-1}} | x_i, y) \approx \sum_{k=1}^{t-1} H(x_k | x_i, y)$

Not intended to be understood at a single slide. Check reference for further understanding.

Feature just random variables

racefox

- $T = argmax_T \left\{ I\{(x_1, x_2, ... x_T), y\} \right\}$ NP-hard direct solution
- Forward Feature Selection $: t_{step=t} = argmax_{i \notin S^{t-1}} \left\{ I(x_{S^{t-1} \cup i} : y) \right\}$
- $I(x_{S^{t-1} \cup i} : y) = I(x_{S^{t-1}} : y) + I(x_i : y | x_{S^{t-1}})$
- $I(x_{S^{t-1} \cup i} : y) = I(x_{S^{t-1}} : y) + I(x_i : y) - I(x_i : x_{S^{t-1}}) + I(x_i : x_{S^{t-1}} | y)$
- $= I(x_{S^{t-1}} : y) + I(x_i : y) - (H(x_{S^{t-1}}) - H(x_{S^{t-1}} | x_i)) + (H(x_{S^{t-1}} | y) - H(x_{S^{t-1}} | x_i, y))$
- $t = argmax_{i \notin S^{t-1}} \left\{ I(x_i : y) + H(x_{S^{t-1}} | x_i) - H(x_{S^{t-1}} | x_i, y) \right\}$
- $H(x_{S^{t-1}} | x_i) \approx \sum_{k=1}^{t-1} H(x_k | x_i)$
- $H(x_{S^{t-1}} | x_i, y) \approx \sum_{k=1}^{t-1} H(x_k | x_i, y)$

Not intended to be understood at a single slide. Check reference for further understanding.

Feature just random variables

racefox

- $T = argmax_T \left\{ I\{(x_1, x_2, ... x_T), y\} \right\}$ NP-hard direct solution
- Forward Feature Selection : $t_{step=t} = argmax_{i \not\in S^{t-1}} \left\{ I(x_{S^{t-1} \cup i} : y) \right\}$
- $I(x_{S^{t-1} \cup i} : y) = I(x_{S^{t-1}} : y) + I(x_i : y | x_{S^{t-1}})$
- $I(x_{S^{t-1} \cup i} : y) = I(x_{S^{t-1}} : y) + I(x_i : y) - I(x_i : x_{S^{t-1}}) + I(x_i : x_{S^{t-1}} | y)$
- $= I(x_{S^{t-1}} : y) + I(x_i : y) - (H(x_{S^{t-1}}) - H(x_{S^{t-1}} | x_i)) + (H(x_{S^{t-1}} | y) - H(x_{S^{t-1}} | x_i, y))$
- $t = argmax_{i \not\in S^{t-1}} \left\{ I(x_i : y) + H(x_{S^{t-1}} | x_i) - H(x_{S^{t-1}} | x_i, y) \right\}$
- $H(x_{S^{t-1}} | x_i) \approx \sum_{k=1}^{t-1} H(x_k | x_i)$
- $H(x_{S^{t-1}} | x_i, y) \approx \sum_{k=1}^{t-1} H(x_k | x_i, y)$

Not intended to be understood at a single slide. Check reference for further understanding.

Feature just random variables

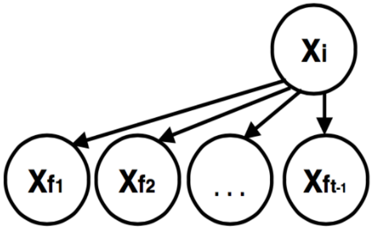
racefox

- $T = argmax_T \left\{ I\{(x_1, x_2, ...x_T), y\} \right\}$NP-hard direct solution
- Forward Feature Selection $: t_{step=t} = argmax_{i \not\in S^{t-1}} \left\{ I(x_{S^{t-1} \cup i} : y) \right\}$
- $I(x_{S^{t-1} \cup i} : y) = I(x_{S^{t-1}} : y) + I(x_i : y | x_{S^{t-1}})$
- $I(x_{S^{t-1} \cup i} : y) = I(x_{S^{t-1}} : y) + I(x_i : y) - I(x_i : x_{S^{t-1}}) + I(x_i : x_{S^{t-1}} | y)$
- $= I(x_{S^{t-1}} : y) + I(x_i : y) - (H(x_{S^{t-1}}) - H(x_{S^{t-1}} | x_i)) + (H(x_{S^{t-1}} | y) - H(x_{S^{t-1}} | x_i, y))$
- $t = argmax_{i \not\in S^{t-1}} \left\{ I(x_i : y) + H(x_{S^{t-1}} | x_i) - H(x_{S^{t-1}} | x_i, y) \right\}$
- $H(x_{S^{t-1}} | x_i) \approx \sum_{k=1}^{t-1} H(x_k | x_i)$
- $H(x_{S^{t-1}} | x_i, y) \approx \sum_{k=1}^{t-1} H(x_k | x_i, y)$

Not intended to be understood at a single slide. Check reference for further understanding.

Feature just random variables

- Assumption1: Feature Independent : $P(x_{St-1}|x_i) = \prod_{k=1}^{t-1} P(x_k|x_i)$
- Assumption2: Class-Conditioned Independent : $P(x_{St-1}|x_i, y) = \prod_{k=1}^{t-1} P(x_k|x_i, y)$
- These have only one common structure fulfillment.
- Contradiction when both are met: $I(X_i : Y) > I(X_1, X_2, \ldots, X_{t-1:Y})$

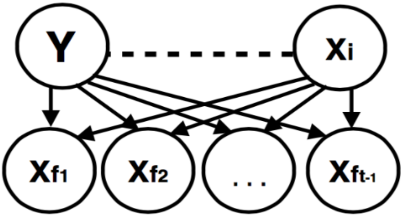


Assumption 1

Feature just random variables

racefox

- Assumption1: Feature Independent $: P(x_{S_{t-1}}|x_i) = \prod_{k=1}^{t-1} P(x_k|x_i)$
- Assumption2: Class-Conditioned Independent $: P(x_{S_{t-1}}|x_i, y) = \prod_{k=1}^{t-1} P(x_k|x_i, y)$
- These have only one common structure fulfillment.
- Contradiction when both are met: $I(X_i : Y) > I(X_1, X_2, \ldots, X_{t-1:Y})$

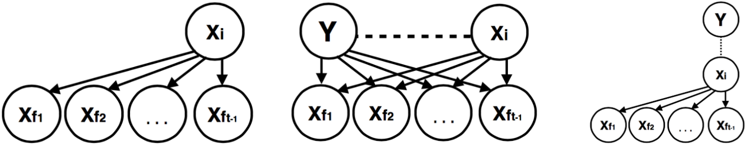


Assumption 2

Feature just random variables

racefox

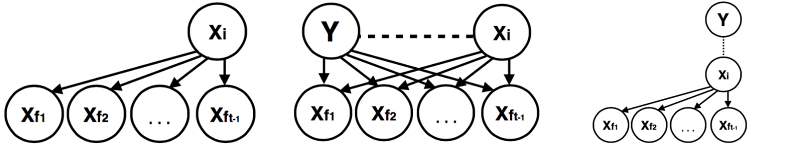
- Assumption1: Feature Independent : $P(x_{St-1}|x_i) = \prod_{k=1}^{t-1} P(x_k|x_i)$
- Assumption2: Class-Conditioned Independent : $P(x_{St-1}|x_i, y) = \prod_{k=1}^{t-1} P(x_k|x_i, y)$
- These have only one common structure fulfillment.
- Contradiction when both are met: $I(X_i : Y) > I(X_1, X_2, ..., X_{t-1:Y})$



Assumption 1 — *Assumption 2* — Satisfying both *Assumption 1* and *Assumption 2*

Feature just random variables

- Assumption1: Feature Independent : $P(x_{S_{t-1}}|x_i) = \prod_{k=1}^{t-1} P(x_k|x_i)$
- Assumption2: Class-Conditioned Independent : $P(x_{S_{t-1}}|x_i, y) = \prod_{k=1}^{t-1} P(x_k|x_i, y)$
- These have only one common structure fulfillment.
- Contradiction when both are met: $I(X_i : Y) > I(X_1, X_2, ..., X_{t-1:Y})$



Assumption 1 *Assumption 2* Satisfying both *Assumption 1* and *Assumption 2*

Feature just random variables

racefox

- $I(x, y) \geq H(x) + < ln[q(x|y)] >_{p(x,y)}$
- $S = argmax_S \{H(x_S) + < ln(q(x_S|y)) >_{p(x_s,y)})\}$
- Swap x with y: $I(x, y) \geq H(y) + < ln[q(y|x)] >)_{p(x,y)} = < ln\left\{ \frac{q(y|x)}{p(y)} \right\} >_{p(x,y)}$
- $S^* = argmax_S \left\{ < ln\{ \frac{q(y|x_S)}{p(y)} \} >_{p(x_s,y)} \right\}$
- $q(y|x_s) = \frac{q(x_s,y)}{q(x_s)} = \frac{q(x_s,y)p(y)}{q(x_s)} = \frac{q(x_s,y)p(y)}{\sum_{y'} q(x_S|y')p(y')}$
- $I(x_s, y) \geq \left\langle ln\left\{ \frac{q(y|x_s)}{q(x_s)} \right\} \right\rangle_{p(x_s,y)} = I_{LB}(x_s : y)$
- $I(x_s, y) - I_{LB}(x_s : y) = \langle KL(p(y|x_s)||q(y|x_s)) \rangle_{p(x_s)}$

Not intended to be understood at a single slide. Check reference for further understanding.

Feature just random variables

racefox

- $I(x, y) \geq H(x) + < ln[q(x|y)] >_{p(x,y)}$
- $S = argmax_S \{H(x_S) + < ln(q(x_S|y)) >_{p(x_s,y)})\}$
- Swap x with y: $I(x, y) \geq H(y) + < ln[q(y|x)] >)_{p(x,y)} =< ln\left\{ \frac{q(y|x)}{p(y)} \right\} >_{p(x,y)}$
- $S^* = argmax_S \left\{ < ln\{ \frac{q(y|x_S)}{p(y)} \} >_{p(x_s,y)} \right\}$
- $q(y|x_s) = \frac{q(x_s,y)}{q(x_s)} = \frac{q(x_s,y)p(y)}{q(x_s)} = \frac{q(x_s,y)p(y)}{\sum_{y'} q(x_S|y')p(y')}$
- $I(x_s, y) \geq \left\langle ln\left\{ \frac{q(y|x_s)}{q(x_s)} \right\} \right\rangle_{p(x_s,y)} = I_{LB}(x_s : y)$
- $I(x_s, y) - I_{LB}(x_s : y) = \langle KL(p(y|x_s)||q(y|x_s)) \rangle_{p(x_s)}$

Not intended to be understood at a single slide. Check reference for further understanding.

Feature just random variables

racefox

- $I(x, y) \geq H(x) + < ln[q(x|y)] >_{p(x,y)}$
- $S = argmax_S \{H(x_S) + < ln(q(x_S|y)) >_{p(x_s,y)}\}$
- Swap x with y: $I(x, y) \geq H(y) + < ln[q(y|x)] >)_{p(x,y)} =< ln\left\{ \frac{q(y|x)}{p(y)} \right\} >_{p(x,y)}$
- $S^* = argmax_S \left\{ < ln\{ \frac{q(y|x_S)}{p(y)} \} >_{p(x_s,y)} \right\}$
- $q(y|x_s) = \frac{q(x_s,y)}{q(x_s)} = \frac{q(x_s,y)p(y)}{q(x_s)} = \frac{q(x_s,y)p(y)}{\sum_{y'} q(x_S|y')p(y')}$
- $I(x_s, y) \geq \left\langle ln\left\{ \frac{q(y|x_s)}{q(x_s)} \right\} \right\rangle_{p(x_s,y)} = I_{LB}(x_s : y)$
- $I(x_s, y) - I_{LB}(x_s : y) = \langle KL(p(y|x_s)||q(y|x_s)) \rangle_{p(x_s)}$

Not intended to be understood at a single slide. Check reference for further understanding.

Feature just random variables

racefox

- $I(x,y) \geq H(x) + < ln[q(x|y)] >_{p(x,y)}$
- $S = argmax_S\{H(x_S) + < ln(q(x_S|y)) >_{p(x_s,y)})\}$
- Swap x with y: $I(x,y) \geq H(y) + < ln[q(y|x)] >)_{p(x,y)} =< ln\left\{\frac{q(y|x)}{p(y)}\right\} >_{p(x,y)}$
- $S^* = argmax_S\left\{ < ln\{\frac{q(y|x_S)}{p(y)}\} >_{p(x_s,y)} \right\}$
- $q(y|x_s) = \frac{q(x_s,y)}{q(x_S)} = \frac{q(x_s,y)p(y)}{q(x_s)} = \frac{q(x_s,y)p(y)}{\sum_{y'} q(x_S|y')p(y')}$
- $I(x_s,y) \geq \left\langle ln\left\{\frac{q(y|x_s)}{q(x_s)}\right\}\right\rangle_{p(x_s,y)} = I_{LB}(x_s:y)$
- $I(x_s,y) - I_{LB}(x_s:y) = \langle KL(p(y|x_s)||q(y|x_s))\rangle_{p(x_s)}$

Not intended to be understood at a single slide. Check reference for further understanding.

Feature just random variables

racefox

- $I(x, y) \geq H(x) + < ln[q(x|y)] >_{p(x,y)}$
- $S = argmax_S\{H(x_S) + < ln(q(x_S|y)) >_{p(x_s,y)})\}$
- Swap x with y: $I(x, y) \geq H(y) + < ln[q(y|x)] >)_{p(x,y)} =< ln\left\{ \frac{q(y|x)}{p(y)} \right\} >_{p(x,y)}$
- $S^* = argmax_S\left\{ < ln\{ \frac{q(y|x_S)}{p(y)} \} >_{p(x_s,y)} \right\}$
- $q(y|x_s) = \frac{q(x_s,y)}{q(x_s)} = \frac{q(x_s,y)p(y)}{q(x_s)} = \frac{q(x_s,y)p(y)}{\sum_{y'} q(x_S|y')p(y')}$
- $I(x_s, y) \geq \left\langle ln\left\{ \frac{q(y|x_s)}{q(x_s)} \right\} \right\rangle_{p(x_s,y)} = I_{LB}(x_s : y)$
- $I(x_s, y) - I_{LB}(x_s : y) = \langle KL(p(y|x_s)||q(y|x_s)) \rangle_{p(x_s)}$

Not intended to be understood at a single slide. Check reference for further understanding.

Feature just random variables

racefox

- $I(x,y) \geq H(x) + < ln[q(x|y)] >_{p(x,y)}$
- $S = argmax_S\{H(x_S) + < ln(q(x_S|y)) >_{p(x_S,y)})\}$
- Swap x with y: $I(x,y) \geq H(y) + < ln[q(y|x)] >)_{p(x,y)} = < ln\left\{\frac{q(y|x)}{p(y)}\right\} >_{p(x,y)}$
- $S^* = argmax_S\left\{ < ln\{\frac{q(y|x_S)}{p(y)}\} >_{p(x_s,y)} \right\}$
- $q(y|x_s) = \frac{q(x_s,y)}{q(x_s)} = \frac{q(x_s,y)p(y)}{q(x_s)} = \frac{q(x_s,y)p(y)}{\sum_{y'} q(x_S|y')p(y')}$
- $I(x_s,y) \geq \left\langle ln\left\{\frac{q(y|x_s)}{q(x_s)}\right\}\right\rangle_{p(x_s,y)} = I_{LB}(x_s : y)$
- $I(x_s,y) - I_{LB}(x_s : y) = \langle KL(p(y|x_s)||q(y|x_s))\rangle_{p(x_s)}$

Not intended to be understood at a single slide. Check reference for further understanding.

Feature just random variables

racefox

- $I(x,y) \geq H(x) + < ln[q(x|y)] >_{p(x,y)}$
- $S = argmax_S\{H(x_S) + < ln(q(x_S|y)) >_{p(x_S,y)})\}$
- Swap x with y: $I(x,y) \geq H(y) + < ln[q(y|x)] >)_{p(x,y)} = < ln\left\{\frac{q(y|x)}{p(y)}\right\} >_{p(x,y)}$
- $S^* = argmax_S\left\{ < ln\{\frac{q(y|x_S)}{p(y)}\} >_{p(x_S,y)} \right\}$
- $q(y|x_s) = \frac{q(x_s,y)}{q(x_s)} = \frac{q(x_s,y)p(y)}{q(x_s)} = \frac{q(x_s,y)p(y)}{\sum_{y'} q(x_S|y')p(y')}$
- $I(x_s,y) \geq \left\langle ln\left\{\frac{q(y|x_s)}{q(x_s)}\right\}\right\rangle_{p(x_s,y)} = I_{LB}(x_s : y)$
- $I(x_s,y) - I_{LB}(x_s : y) = \langle KL(p(y|x_s)||q(y|x_s))\rangle_{p(x_s)}$

Not intended to be understood at a single slide. Check reference for further understanding.

Variational feature selection under auto-regressive decomposition

- $q(x_s|y) = q(x_1|y)\prod_{t=2}^{T} q(x_t|x_{T\geq t}, y)$
- $I_{LB}(x_s : y) = \frac{1}{N}\sum_{x^k, y^k} ln \frac{\hat{q}(x_S^k|y^k)}{\hat{q}(x_S^{(k)})}$

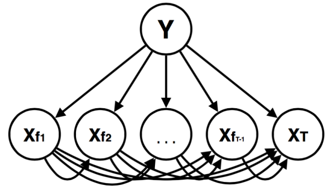


Figure 2: Auto-regressive decomposition for $q(\mathbf{x}_S|\mathbf{y})$

- MI assesses the informativeness of features
- It requires a lot of observation if the dimensionality of the data is very high

Variational feature selection under auto-regressive decomposition

- $q(x_s|y) = q(x_1|y)\prod_{t=2}^{T} q(x_t|x_{T\geq t}, y)$
- $I_{LB}(x_s : y) = \frac{1}{N}\sum_{x^k,y^k} ln \frac{\hat{q}(x_S^k|y^k)}{\hat{q}(x_S^{(k)})}$

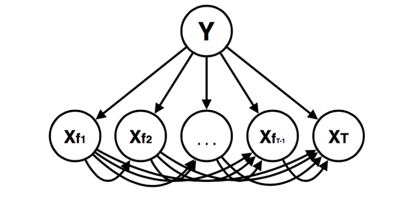


Figure 2: Auto-regressive decomposition for $q(\mathbf{x}_S|\mathbf{y})$

- MI assesses the informativeness of features
- It requires a lot of observation if the dimensionality of the data is very high

Variational feature selection under auto-regressive decomposition

- $q(x_s|y) = q(x_1|y)\prod_{t=2}^{T} q(x_t|x_{T\geq t}, y)$
- $I_{LB}(x_s : y) = \frac{1}{N}\sum_{x^k, y^k} ln \frac{\hat{q}(x_S^k|y^k)}{\hat{q}(x_S^{(k)})}$

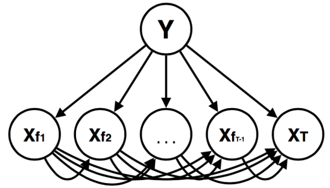


Figure 2: Auto-regressive decomposition for $q(\mathbf{x}_S|\mathbf{y})$

- MI assesses the informativeness of features
- It requires a lot of observation if the dimensionality of the data is very high

Variational feature selection under auto-regressive decomposition

racefox

- $q(x_s|y) = q(x_1|y)\prod_{t=2}^{T} q(x_t|x_{T\geq t}, y)$
- $I_{LB}(x_s : y) = \frac{1}{N}\sum_{x^k, y^k} ln \frac{\hat{q}(x_S^k|y^k)}{\hat{q}(x_S^{(k)})}$

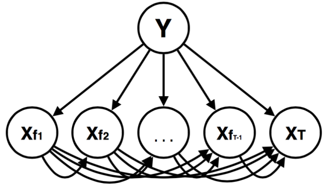


Figure 2: Auto-regressive decomposition for $q(\mathbf{x}_S|\mathbf{y})$

- MI assesses the informativeness of features
- It requires a lot of observation if the dimensionality of the data is very high

Kernel Feature Selection via Conditional Covariance Minimization
Chen et al

Feature selections prediction perspective

racefox

- F is a class of functions from X to Y.
- L is a **loss function** defined by the user (MSE).
- Prediction error: $\epsilon_F = inf_{f \in F} E_{X,Y} L(Y, f(X))$
- Solve the problem:

$$min_{T:|T| \leq m} \epsilon_F(X_T) = min_{T:|T| \leq m} inf_{f \in F} E_{X,Y} \left\{ L(T, f(X)) \right\}$$

Feature selections prediction perspective

racefox

- F is a class of functions from X to Y.
- L is a **loss function** defined by the user (MSE).
- Prediction error: $\epsilon_F = inf_{f \in F} E_{X,Y} L(Y, f(X))$
- Solve the problem:

  $min_{T:|T| \leq m} \epsilon_F(X_T) = min_{T:|T| \leq m} inf_{f \in F} E_{X,Y} \left\{ L(T, f(X)) \right\}$

Feature selections prediction perspective

- F is a class of functions from X to Y.
- L is a **loss function** defined by the user (MSE).
- Prediction error: $\epsilon_F = inf_{f \in F} E_{X,Y} L(Y, f(X))$
- Solve the problem:

$$min_{T:|T| \leq m} \epsilon_F(X_T) = min_{T:|T| \leq m} inf_{f \in F} E_{X,Y} \left\{ L(T, f(X)) \right\}$$

Feature selections prediction perspective

- F is a class of functions from X to Y.
- L is a **loss function** defined by the user (MSE).
- Prediction error: $\epsilon_F = inf_{f \in F} E_{X,Y} L(Y, f(X))$
- Solve the problem:

  $$min_{T:|T| \leq m} \epsilon_F(X_T) = min_{T:|T| \leq m} inf_{f \in F} E_{X,Y} \left\{ L(T, f(X)) \right\}$$

Correlation vs Association

racefox

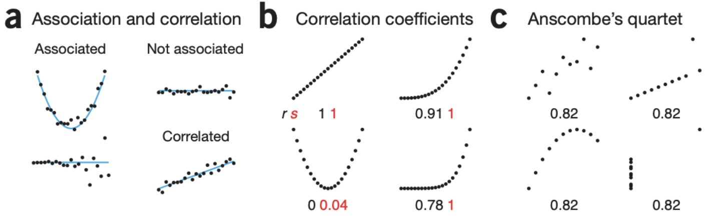
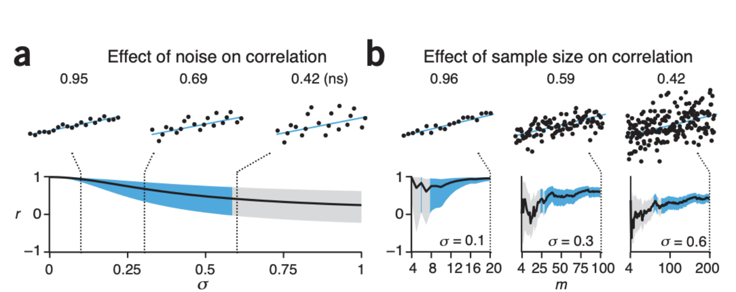


Figure 1 | Correlation is a type of association and measures increasing or decreasing trends quantified using correlation coefficients. (**a**) Scatter plots of associated (but not correlated), non-associated and correlated variables. In the lower association example, variance in *y* is increasing with *x*. (**b**) The Pearson correlation coefficient (*r*, black) measures linear trends, and the Spearman correlation coefficient (*s*, red) measures increasing or decreasing trends. (**c**) Very different data sets may have similar *r* values. Descriptors such as curvature or the presence of outliers can be more specific.

Altman et al

- Correlation: How much variance is explained → $\rho_{X,Y} = \frac{cov(X,Y)}{\sigma_X * \sigma_Y}$
- Covariance: X and Y co-vary, → $cov(X,Y) = \rho_{X,Y} * \sigma_X * \sigma_Y$

Correlation vs Association



Altman et al

- Correlation: How much variance is explained $\rightarrow \rho_{X,Y} = \frac{cov(X,Y)}{\sigma_X * \sigma_Y}$
- Covariance: X and Y co-vary, $\rightarrow cov(X,Y) = \rho_{X,Y} * \sigma_X * \sigma_Y$

Correlation vs Association

racefox

Altman et al

- Correlation: How much variance is explained $\rightarrow \rho_{X,Y} = \frac{cov(X,Y)}{\sigma_X * \sigma_Y}$
- Covariance: X and Y co-vary, $\rightarrow cov(X,Y) = \rho_{X,Y} * \sigma_X * \sigma_Y$

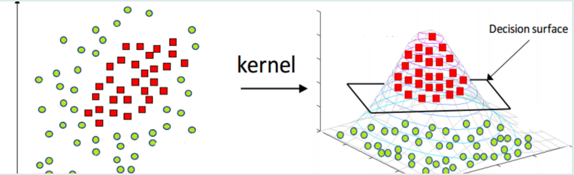
Correlation vs Association

Altman et al

- Correlation: How much variance is explained $\rightarrow \rho_{X,Y} = \frac{cov(X,Y)}{\sigma_X * \sigma_Y}$
- Covariance: X and Y co-vary, $\rightarrow cov(X, Y) = \rho_{X,Y} * \sigma_X * \sigma_Y$

Kernel trick

racefox



$$k(x_i, x_j) = \varphi(x_i)^T \varphi(x_j) \tag{3}$$

$$\varphi : x \to \varphi(x) \tag{4}$$

Conditional Covariance Operator(CCO)

- CCO computes a measure of the conditional dependency for random variables.
- (Hx,Kx) and (Hy,Ky) the reproducible kernel Hilbert space (RKHSs) of functions of X and Y respectively.
- (X,Y) a random array on (XxY) with distribution P(X,Y)
- The cross-covariance operator associated with the pair (X,Y) is the mapping $\sum_{X,Y} : H_X - > H_Y$
- s.t: $E_{X,Y}\{(f(X) - E_X[f(X)])(g(Y) - E_Y[g(Y)])\} : \forall g \in H_Y, f \in H_X$

Conditional Covariance Operator(CCO)

- CCO computes a measure of the conditional dependency for random variables.
- (Hx,Kx) and (Hy,Ky) the reproducible kernel Hilbert space (RKHSs) of functions of X and Y respectively.
- (X,Y) a random array on ($X x Y$) with distribution P(X,Y)
- The cross-covariance operator associated with the pair (X,Y) is the mapping $\sum_{X,Y} : H_X -> H_Y$
- s.t: $E_{X,Y}\{(f(X) - E_X[f(X)])(g(Y) - E_Y[g(Y)])\}$: $\forall g \in H_Y, f \in H_X$

Conditional Covariance Operator(CCO)

- CCO computes a measure of the conditional dependency for random variables.
- (Hx,Kx) and (Hy,Ky) the reproducible kernel Hilbert space (RKHSs) of functions of X and Y respectively.
- (X,Y) a random array on ($X x Y$) with distribution P(X,Y)
- The cross-covariance operator associated with the pair (X,Y) is the mapping $\sum_{X,Y} : H_X -> H_Y$
- s.t: $E_{X,Y}\{(f(X) - E_X[f(X)])(g(Y) - E_Y[g(Y)])\}$: $\forall g \in H_Y, f \in H_X$

Conditional Covariance Operator(CCO)

racefox

- CCO computes a measure of the conditional dependency for random variables.
- (Hx,Kx) and (Hy,Ky) the reproducible kernel Hilbert space (RKHSs) of functions of X and Y respectively.
- (X,Y) a random array on $(X x Y)$ with distribution P(X,Y)
- The cross-covariance operator associated with the pair (X,Y) is the mapping $\sum_{X,Y} : H_X - > H_Y$
- s.t: $E_{X,Y}\{(f(X) - E_X[f(X)])(g(Y) - E_Y[g(Y)])\}$: $\forall g \in H_Y, f \in H_X$

Conditional Covariance Operator(CCO)

racefox

- CCO computes a measure of the conditional dependency for random variables.
- (Hx,Kx) and (Hy,Ky) the reproducible kernel Hilbert space (RKHSs) of functions of X and Y respectively.
- (X,Y) a random array on $(X x Y)$ with distribution P(X,Y)
- The cross-covariance operator associated with the pair (X,Y) is the mapping $\sum_{X,Y} : H_X -> H_Y$
- s.t: $E_{X,Y}\{(f(X) - E_X[f(X)])(g(Y) - E_Y[g(Y)])\}$: $\forall g \in H_Y, f \in H_X$

Conditional Covariance Operator(CCO)

racefox

- There exists unique bounded operator $V_{Y,X}$,s.t:
- $< g, \sum_{YX} f >_{H_y} = \sum_{YX} = (\sum_{YY})^{1/2} V_{YX} (\sum_{XX})^{1/2}$
- CCO:$\sum_{XX|Y} = \sum_{YY} - (\sum_{YY})^{1/2} V_{YX} V_{XY} (\sum_{XX})^{1/2}$
- CCO captures the conditional variance of Y given X
- $L^2(P_X)$ is the space of all square-integrable[1] functions on X
- If $H_X + R$ is dense in $L^2(P_X)$
- $< g, \sum_{XX|X} g >_{H_Y} = E[var_{Y|X}[g(Y)|X]], \forall g \in H_Y$
- The residual error of $g(Y)$ (where Y is part of Hy) can be characterized by the CCO
- $< g, \sum_{YY|X} g >_{H_y} = inf_{f \in H_x} E_{X,Y}\{(g(Y) - E_Y[g(Y)]) - (f(X) - E_X[f(X)])\}$

Not intended to be understood at a single slide. Check reference for further understanding.

[1] $\int_{-\infty}^{+\infty} |f(x)|^2 dx < \infty$

Conditional Covariance Operator(CCO)

- There exists unique bounded operator $V_{Y,X}$,s.t:
- $< g, \sum_{YX} f >_{H_y} = \sum_{YX} = (\sum_{YY})^{1/2} V_{YX} (\sum_{XX})^{1/2}$
- CCO:$\sum_{XX|Y} = \sum_{YY} - (\sum_{YY})^{1/2} V_{YX} V_{XY} (\sum_{XX})^{1/2}$
- CCO captures the conditional variance of Y given X
- $L^2(P_X)$ is the space of all square-integrable[1] functions on X
- If $H_X + R$ is dense in $L^2(P_X)$
- $< g, \sum_{XX|X} g >_{H_Y} = E[var_{Y|X}[g(Y)|X]], \forall g \in H_Y$
- The residual error of $g(Y)$ (where Y is part of Hy) can be characterized by the CCO
- $< g, \sum_{YY|X} g >_{H_y} = inf_{f \in H_x} E_{X,Y}\{(g(Y) - E_Y[g(Y)]) - (f(X) - E_X[f(X)])\}$

Not intended to be understood at a single slide. Check reference for further understanding.

[1] $\int_{-\infty}^{+\infty} |f(x)|^2 dx < \infty$

Conditional Covariance Operator(CCO)

- There exists unique bounded operator $V_{Y,X}$,s.t:
- $< g, \sum_{YX} f >_{H_y} = \sum_{YX} = (\sum_{YY})^{1/2} V_{YX} (\sum_{XX})^{1/2}$
- CCO:$\sum_{XX|Y} = \sum_{YY} - (\sum_{YY})^{1/2} V_{YX} V_{XY} (\sum_{XX})^{1/2}$
- CCO captures the conditional variance of Y given X
- $L^2(P_X)$ is the space of all square-integrable[1] functions on X
- If $H_X + R$ is dense in $L^2(P_X)$
- $< g, \sum_{XX|X} g >_{H_Y} = E[var_{Y|X}[g(Y)|X]], \forall g \in H_Y$
- The residual error of $g(Y)$ (where Y is part of Hy) can be characterized by the CCO
- $< g, \sum_{YY|X} g >_{H_y} = inf_{f \in H_x} E_{X,Y}\{(g(Y) - E_Y[g(Y)]) - (f(X) - E_X[f(X)])\}$

Not intended to be understood at a single slide. Check reference for further understanding.

[1] $\int_{-\infty}^{+\infty} |f(x)|^2 dx < \infty$

Conditional Covariance Operator(CCO)

racefox

- There exists unique bounded operator $V_{Y,X}$,s.t:
- $< g, \sum_{YX} f >_{H_y} = \sum_{YX} = (\sum_{YY})^{1/2} V_{YX} (\sum_{XX})^{1/2}$
- CCO:$\sum_{XX|Y} = \sum_{YY} - (\sum_{YY})^{1/2} V_{YX} V_{XY} (\sum_{XX})^{1/2}$
- CCO captures the conditional variance of Y given X
- $L^2(P_X)$ is the space of all square-integrable[1] functions on X
- If $H_X + R$ is dense in $L^2(P_X)$
- $< g, \sum_{XX|X} g >_{H_Y} = E[var_{Y|X}[g(Y)|X]], \forall g \in H_Y$
- The residual error of $g(Y)$ (where Y is part of Hy) can be characterized by the CCO
- $< g, \sum_{YY|X} g >_{H_y} = inf_{f \in H_x} E_{X,Y}\{(g(Y) - E_Y[g(Y)]) - (f(X) - E_X[f(X)])\}$

Not intended to be understood at a single slide. Check reference for further understanding.

[1] $\int_{-\infty}^{+\infty} |f(x)|^2 dx < \infty$

Conditional Covariance Operator(CCO)

- There exists unique bounded operator $V_{Y,X}$,s.t:
- $< g, \sum_{YX} f >_{H_y} = \sum_{YX} = (\sum_{YY})^{1/2} V_{YX} (\sum_{XX})^{1/2}$
- CCO:$\sum_{XX|Y} = \sum_{YY} - (\sum_{YY})^{1/2} V_{YX} V_{XY} (\sum_{XX})^{1/2}$
- CCO captures the conditional variance of Y given X
- $L^2(P_X)$ is the space of all square-integrable[1] functions on X
- If $H_X + R$ is dense in $L^2(P_X)$
- $< g, \sum_{XX|X} g >_{H_Y} = E[var_{Y|X}[g(Y)|X]], \forall g \in H_Y$
- The residual error of $g(Y)$ (where Y is part of Hy) can be characterized by the CCO
- $< g, \sum_{YY|X} g >_{H_y} = inf_{f \in H_x} E_{X,Y}\{(g(Y) - E_Y[g(Y)]) - (f(X) - E_X[f(X)])\}$

Not intended to be understood at a single slide. Check reference for further understanding.

[1] $\int_{-\infty}^{+\infty} |f(x)|^2 dx < \infty$

Conditional Covariance Operator(CCO)

racefox

- There exists unique bounded operator $V_{Y,X}$,s.t:
- $< g, \sum_{YX} f >_{H_y} = \sum_{YX} = (\sum_{YY})^{1/2} V_{YX} (\sum_{XX})^{1/2}$
- CCO:$\sum_{XX|Y} = \sum_{YY} - (\sum_{YY})^{1/2} V_{YX} V_{XY} (\sum_{XX})^{1/2}$
- CCO captures the conditional variance of Y given X
- $L^2(P_X)$ is the space of all square-integrable[1] functions on X
- If $H_X + R$ is dense in $L^2(P_X)$
- $< g, \sum_{XX|X} g >_{H_Y} = E[var_{Y|X}[g(Y)|X]], \forall g \in H_Y$
- The residual error of $g(Y)$ (where Y is part of Hy) can be characterized by the CCO
- $< g, \sum_{YY|X} g >_{H_y} = inf_{f \in H_x} E_{X,Y}\{(g(Y) - E_Y[g(Y)]) - (f(X) - E_X[f(X)])\}$

Not intended to be understood at a single slide. Check reference for further understanding.

[1] $\int_{-\infty}^{+\infty} |f(x)|^2 dx < \infty$

Conditional Covariance Operator(CCO)

- There exists unique bounded operator $V_{Y,X}$,s.t:
- $< g, \sum_{YX} f >_{H_y} = \sum_{YX} = (\sum_{YY})^{1/2} V_{YX} (\sum_{XX})^{1/2}$
- CCO:$\sum_{XX|Y} = \sum_{YY} - (\sum_{YY})^{1/2} V_{YX} V_{XY} (\sum_{XX})^{1/2}$
- CCO captures the conditional variance of Y given X
- $L^2(P_X)$ is the space of all square-integrable[1] functions on X
- If $H_X + R$ is dense in $L^2(P_X)$
- $< g, \sum_{XX|X} g >_{H_Y} = E[var_{Y|X}[g(Y)|X]], \forall g \in H_Y$
- The residual error of $g(Y)$ (where Y is part of Hy) can be characterized by the CCO
- $< g, \sum_{YY|X} g >_{H_y} = inf_{f \in H_x} E_{X,Y}\{(g(Y) - E_Y[g(Y)]) - (f(X) - E_X[f(X)])\}$

Not intended to be understood at a single slide. Check reference for further understanding.

[1] $\int_{-\infty}^{+\infty} |f(x)|^2 dx < \infty$

Conditional Covariance Operator(CCO)

- There exists unique bounded operator $V_{Y,X}$,s.t:
- $< g, \sum_{YX} f >_{H_y} = \sum_{YX} = (\sum_{YY})^{1/2} V_{YX} (\sum_{XX})^{1/2}$
- CCO:$\sum_{XX|Y} = \sum_{YY} - (\sum_{YY})^{1/2} V_{YX} V_{XY} (\sum_{XX})^{1/2}$
- CCO captures the conditional variance of Y given X
- $L^2(P_X)$ is the space of all square-integrable[1] functions on X
- If $H_X + R$ is dense in $L^2(P_X)$
- $< g, \sum_{XX|X} g >_{H_Y} = E[var_{Y|X}[g(Y)|X]], \forall g \in H_Y$
- The residual error of $g(Y)$ (where Y is part of Hy) can be characterized by the CCO
- $< g, \sum_{YY|X} g >_{H_y} =$ $inf_{f \in H_x} E_{X,Y}\{(g(Y) - E_Y[g(Y)]) - (f(X) - E_X[f(X)])\}$

Not intended to be understood at a single slide. Check reference for further understanding.

[1] $\int_{-\infty}^{+\infty} |f(x)|^2 dx < \infty$

Conditional Covariance Operator(CCO)

racefox

- There exists unique bounded operator $V_{Y,X}$,s.t:
- $< g, \sum_{YX} f >_{H_y} = \sum_{YX} = (\sum_{YY})^{1/2} V_{YX} (\sum_{XX})^{1/2}$
- CCO:$\sum_{XX|Y} = \sum_{YY} - (\sum_{YY})^{1/2} V_{YX} V_{XY} (\sum_{XX})^{1/2}$
- CCO captures the conditional variance of Y given X
- $L^2(P_X)$ is the space of all square-integrable[1] functions on X
- If $H_X + R$ is dense in $L^2(P_X)$
- $< g, \sum_{XX|X} g >_{H_Y} = E[var_{Y|X}[g(Y)|X]], \forall g \in H_Y$
- The residual error of $g(Y)$ (where Y is part of Hy) can be characterized by the CCO
- $< g, \sum_{YY|X} g >_{H_y} =$ $inf_{f \in H_x} E_{X.Y} \{(g(Y) - E_Y[g(Y)]) - (f(X) - E_X[f(X)])\}$

Not intended to be understood at a single slide. Check reference for further understanding.

[1] $\int_{-\infty}^{+\infty} |f(x)|^2 dx < \infty$

Proposed method

- Let (H_1, k_1) be the RKHS $X \subset R^d$
- Let $T \subseteq [d]$ be a subset of features with cardinality $m \leq d$
- We define $k_1^T(x, \tilde{x}) = k_1(x^T, \tilde{x}^T) \forall x, \tilde{x} \in X$
- k_1 is permutation(π) invariance
  $\forall x, \tilde{x} \in X, k_1(x, \tilde{x}) = k_1(x_\pi, \tilde{x_\pi})$[2]
- $trace[\sum_{XX|Y}]$[3] interpreted as a dependency measure.
- (H,k) is characteristic if $P \rightarrow E_P[k(X, :)]$ is 1to1 map.
- If k is bounded$\rightarrow$ $H + R$ is dense in $L^2(P), \forall P$.

---

[2] $(x_{\pi_1}, x_{\pi_2}, ..x_{\pi_d}) as, x_\pi$

[3] $trace[A_{(N \times N)}] = \sum_{i=1}^{N} A_{(i,i)}$

Proposed method

racefox

- Let (H_1, k_1) be the RKHS $X \subset R^d$
- Let $T \subseteq [d]$ be a subset of features with cardinality $m \leq d$
- We define $k_1^T(x, \tilde{x}) = k_1(x^T, \tilde{x}^T) \forall x, \tilde{x} \in X$
- k_1 is permutation(π) invariance
  $\forall x, \tilde{x} \in X, k_1(x, \tilde{x}) = k_1(x_\pi, \tilde{x_\pi})$[2]
- $trace[\sum_{XX|Y}]$[3] interpreted as a dependency measure.
- (H,k) is characteristic if $P \rightarrow E_P[k(X, :)]$ is 1to1 map.
- If k is bounded$\rightarrow$ $H + R$ is dense in $L^2(P), \forall P$.

---

[2] $(x_{\pi_1}, x_{\pi_2}, ..x_{\pi_d}) as, x_\pi$
[3] $trace[A_{(N \times N)}] = \sum_{i=1}^{N} A_{(i,i)}$

Proposed method

racefox

- Let (H_1, k_1) be the RKHS $X \subset R^d$
- Let $T \subseteq [d]$ be a subset of features with cardinality $m \leq d$
- We define $k_1^T(x, \tilde{x}) = k_1(x^T, \tilde{x}^T) \forall x, \tilde{x} \in X$
- k_1 is permutation(π) invariance $\forall x, \tilde{x} \in X, k_1(x, \tilde{x}) = k_1(x_\pi, \tilde{x_\pi})$[2]
- $trace[\sum_{XX|Y}]$[3] interpreted as a dependency measure.
- (H,k) is characteristic if $P \rightarrow E_P[k(X, :)]$ is 1to1 map.
- If k is bounded$\rightarrow$ $H + R$ is dense in $L^2(P), \forall P$.

[2] $(x_{\pi_1}, x_{\pi_2}, ..x_{\pi_d}) as, x_\pi$

[3] $trace[A_{(N \times N)}] = \sum_{i=1}^{N} A_{(i,i)}$

Proposed method

- Let (H_1, k_1) be the RKHS $X \subset R^d$
- Let $T \subseteq [d]$ be a subset of features with cardinality $m \leq d$
- We define $k_1^T(x, \tilde{x}) = k_1(x^T, \tilde{x}^T) \forall x, \tilde{x} \in X$
- k_1 is permutation(π) invariance
  $\forall x, \tilde{x} \in X, k_1(x, \tilde{x}) = k_1(x_\pi, \tilde{x_\pi})$[2]
- $trace[\sum_{XX|Y}]$[3] interpreted as a dependency measure.
- (H,k) is characteristic if $P \to E_P[k(X, :)]$ is 1to1 map.
- If k is bounded$\to$ $H + R$ is dense in $L^2(P), \forall P$.

[2] $(x_{\pi_1}, x_{\pi_2}, ..x_{\pi_d}) as, x_\pi$

[3] $trace[A_{(N \times N)}] = \sum_{i=1}^{N} A_{(i,i)}$

Proposed method

racefox

- Let (H_1, k_1) be the RKHS $X \subset R^d$
- Let $T \subseteq [d]$ be a subset of features with cardinality $m \leq d$
- We define $k_1^T(x, \tilde{x}) = k_1(x^T, \tilde{x}^T) \forall x, \tilde{x} \in X$
- k_1 is permutation(π) invariance $\forall x, \tilde{x} \in X, k_1(x, \tilde{x}) = k_1(x_\pi, \tilde{x_\pi})$[2]
- $trace[\sum_{XX|Y}]$[3] interpreted as a dependency measure.
- (H,k) is characteristic if $P \to E_P[k(X, :)]$ is 1to1 map.
- If k is bounded$\to H + R$ is dense in $L^2(P), \forall P$.

[2] $(x_{\pi_1}, x_{\pi_2}, ..x_{\pi_d}) as, x_\pi$
[3] $trace[A_{(N \times N)}] = \sum_{i=1}^{N} A_{(i,i)}$

Proposed method

racefox

- Let (H_1, k_1) be the RKHS $X \subset R^d$
- Let $T \subseteq [d]$ be a subset of features with cardinality $m \leq d$
- We define $k_1^T(x, \tilde{x}) = k_1(x^T, \tilde{x}^T) \forall x, \tilde{x} \in X$
- k_1 is permutation(π) invariance
  $\forall x, \tilde{x} \in X, k_1(x, \tilde{x}) = k_1(x_\pi, \tilde{x_\pi})$[2]
- $trace[\sum_{XX|Y}]$[3] interpreted as a dependency measure.
- (H,k) is characteristic if $P \rightarrow E_P[k(X, :)]$ is 1to1 map.
- If k is bounded$\rightarrow$ $H + R$ is dense in $L^2(P), \forall P$.

[2] $(x_{\pi_1}, x_{\pi_2}, ..x_{\pi_d}) as, x_\pi$
[3] $trace[A_{(N \times N)}] = \sum_{i=1}^{N} A_{(i,i)}$

Proposed method

racefox

- Let (H_1, k_1) be the RKHS $X \subset R^d$
- Let $T \subseteq [d]$ be a subset of features with cardinality $m \leq d$
- We define $k_1^T(x, \tilde{x}) = k_1(x^T, \tilde{x}^T) \forall x, \tilde{x} \in X$
- k_1 is permutation(π) invariance
  $\forall x, \tilde{x} \in X, k_1(x, \tilde{x}) = k_1(x_\pi, \tilde{x_\pi})$[2]
- $trace[\sum_{XX|Y}]$[3] interpreted as a dependency measure.
- (H,k) is characteristic if $P \rightarrow E_P[k(X, :)]$ is 1to1 map.
- If k is bounded$\rightarrow$ $H + R$ is dense in $L^2(P), \forall P$.

[2] $(x_{\pi_1}, x_{\pi_2}, ..x_{\pi_d}) as, x_\pi$
[3] $trace[A_{(N \times N)}] = \sum_{i=1}^{N} A_{(i,i)}$

Proposed method

racefox

- **L1**[4]: if k_1 is bounded and characteristic$\rightarrow$ $\tilde{k_1}$ is characteristic
- **TH2**[5]: if $(H_1, k_1) and (H_2, k_2)$ are characteristic:
  $\sum_{YY|X} \leq \sum_{YY|X_T}$
  $\sum_{YY|X} = \sum_{YY|X_T} : iff : Y \perp X|X_T$
- **C3**[6]:If (H_1, k_1) is characteristic,
  $\{y \in [0,1] : where \sum_i y_i = 1\} \subset R_k$, and (H_2, k_2) includes the identity function on Y, then we have:
  $Tr(\sum_{YY|X}) \leq Tr(\sum_{YY|X_T}), \forall T$
  $Tr(\sum_{YY|X}) = Tr(\sum_{YY|X_T}) : iff : Y \perp X|X_T$
- Univariate Objective:$min_{T:|T|=m} Q(T) = Tr(\sum_{YY|X_T})$

[4] L$\rightarrow$Lemma

[5] TH$\rightarrow$ *Theorem*

[6] C$\rightarrow$Corollari

Proposed method

racefox

- **L1**[4]: if k_1 is bounded and characteristic$\rightarrow \tilde{k_1}$ is characteristic
- **TH2**[5]: if $(H_1, k_1) and (H_2, k_2)$ are characteristic:
  $\sum_{YY|X} \leq \sum_{YY|X_T}$
  $\sum_{YY|X} = \sum_{YY|X_T} : iff : Y \perp X|X_T$
- **C3**[6]:If (H_1, k_1) is characteristic, $\{y \in [0,1] : where \sum_i y_i = 1\} \subset R_k$, and (H_2, k_2) includes the identity function on Y, then we have:
  $Tr(\sum_{YY|X}) \leq Tr(\sum_{YY|X_T}), \forall T$
  $Tr(\sum_{YY|X}) = Tr(\sum_{YY|X_T}) : iff : Y \perp X|X_T$
- Univariate Objective:$min_{T:|T|=m} Q(T) = Tr(\sum_{YY|X_T})$

[4] L$\rightarrow$Lemma
[5] TH$\rightarrow$ *Theorem*
[6] C$\rightarrow$Corollari

Proposed method

- **L1**[4]: if k_1 is bounded and characteristic$\rightarrow \tilde{k_1}$ is characteristic
- **TH2**[5]: if $(H_1, k_1) and (H_2, k_2)$ are characteristic:
  $\sum_{YY|X} \leq \sum_{YY|X_T}$
  $\sum_{YY|X} = \sum_{YY|X_T} : iff : Y \perp X|X_T$
- **C3**[6]:If (H_1, k_1) is characteristic,
  $\{y \in [0,1] : where \sum_i y_i = 1\} \subset R_k$, and (H_2, k_2) includes the identity function on Y, then we have:
  $Tr(\sum_{YY|X}) \leq Tr(\sum_{YY|X_T}), \forall T$
  $Tr(\sum_{YY|X}) = Tr(\sum_{YY|X_T}) : iff : Y \perp X|X_T$
- Univariate Objective:$min_{T:|T|=m} Q(T) = Tr(\sum_{YY|X_T})$

[4] L$\rightarrow$Lemma

[5] TH$\rightarrow$ *Theorem*

[6] C$\rightarrow$Corollari

Proposed method

- **L1**[4]: if k_1 is bounded and characteristic$\rightarrow \tilde{k_1}$ is characteristic
- **TH2**[5]: if $(H_1, k_1) and (H_2, k_2)$ are characteristic:
  $\sum_{YY|X} \leq \sum_{YY|X_T}$
  $\sum_{YY|X} = \sum_{YY|X_T} : iff : Y \perp X|X_T$
- **C3**[6]:If (H_1, k_1) is characteristic,
  $\{y \in [0, 1] : where \sum_i y_i = 1\} \subset R_k$, and (H_2, k_2) includes the identity function on Y, then we have:
  $Tr(\sum_{YY|X}) \leq Tr(\sum_{YY|X_T}), \forall T$
  $Tr(\sum_{YY|X}) = Tr(\sum_{YY|X_T}) : iff : Y \perp X|X_T$
- Univariate Objective:$min_{T:|T|=m} Q(T) = Tr(\sum_{YY|X_T})$

[4] L$\rightarrow$Lemma
[5] TH$\rightarrow$ *Theorem*
[6] C$\rightarrow$Corollari

Proposed method

racefox

- **L1**[4]: if k_1 is bounded and characteristic$\rightarrow$ $\tilde{k_1}$ is characteristic
- **TH2**[5]: if $(H_1, k_1) and (H_2, k_2)$ are characteristic:
  $\sum_{YY|X} \leq \sum_{YY|X_T}$
  $\sum_{YY|X} = \sum_{YY|X_T}$: *iff* : $Y \perp X|X_T$
- **C3**[6]:If (H_1, k_1) is characteristic,
  $\{y \in [0,1] : where \sum_i y_i = 1\} \subset R_k$, and (H_2, k_2) includes the identity function on Y, then we have:
  $Tr(\sum_{YY|X}) \leq Tr(\sum_{YY|X_T}), \forall T$
  $Tr(\sum_{YY|X}) = Tr(\sum_{YY|X_T})$: *iff* : $Y \perp X|X_T$
- Univariate Objective: $min_{T:|T|=m} Q(T) = Tr(\sum_{YY|X_T})$

[4] L$\rightarrow$Lemma
[5] TH$\rightarrow$ *Theorem*
[6] C$\rightarrow$Corollari

Proposed method

racefox

- **L1**[4]: if k_1 is bounded and characteristic$\rightarrow \tilde{k_1}$ is characteristic
- **TH2**[5]: if $(H_1, k_1) and (H_2, k_2)$ are characteristic:
  $\sum_{YY|X} \leq \sum_{YY|X_T}$
  $\sum_{YY|X} = \sum_{YY|X_T} : iff : Y \perp X|X_T$
- **C3**[6]:If (H_1, k_1) is characteristic,
  $\{y \in [0, 1] : where \sum_i y_i = 1\} \subset R_k$, and (H_2, k_2) includes the identity function on Y, then we have:
  $Tr(\sum_{YY|X}) \leq Tr(\sum_{YY|X_T}), \forall T$
  $Tr(\sum_{YY|X}) = Tr(\sum_{YY|X_T}) : iff : Y \perp X|X_T$
- Univariate Objective:$min_{T:|T|=m} Q(T) = Tr(\sum_{YY|X_T})$

[4] L$\rightarrow$Lemma
[5] TH$\rightarrow$ *Theorem*
[6] C$\rightarrow$Corollari

Proposed method

racefox

- **C4**:Let: $\sum_{YY|X_T}$denote CCO of (X_T, Y)in $(\tilde{H_1}, \tilde{k_1})$
  denote: $F_m = \tilde{H_1} + R = f + c : f \in \tilde{H_1}, c \in R$
  then: $Tr(\sum_{YY|X_T}) = \epsilon_{F_m}(X_T) = inf_{f \in F} E_{X,Y}(Y - f(X_T))^2$
- Given n samples $(x_1, y_1), (x_2, y_2), ...(x_n, y_n)$ the empirical estimate is given by:
  $Tr(\sum_{Y\tilde{Y}|X_T}{}^{(n)}) = trace\left\{\sum_{\tilde{YY}}{}^{(n)} - \sum_{\tilde{Y},X_T}{}^{(n)}[\sum_{\tilde{X_T},Y_T}{}^{(n)} + \epsilon I]\sum_{\tilde{X_T}Y}{}^{(n)}\right\}$
  $Tr(\sum_{Y\tilde{Y}|X_T}{}^{(n)}) = \epsilon_n trace\{G_Y[G_X + n\epsilon_n I_n]\}$
  where $G_X = (I_n - \frac{1}{n} I * I^T) K_{X_T} (I_n - \frac{1}{n} I * I^T)$
  and $G_Y = (I_n - \frac{1}{n} I * I^T) K_{Y_T} (I_n - \frac{1}{n} I * I^T)$
- WLG: $trace[G_Y(G_{X_T} + n\epsilon_n I_n)^{-1}] = trace[YY^T(G_{X_T} + n\epsilon_N I_N)^{-1}] = trace[Y^T(G_{X_T} + n\epsilon_N I_N)^{-1} Y]$
- Univariate Objective:$min_{|T|=m} \tilde{Q^{(n)}} = Y^T(G_{X_T} + n\epsilon_N I_N)^{-1} Y$
  where $Y = (y_1, y_2, ..., y_n)$ is a n-dimensional vector.

Not intended to be understood at a single slide. Check reference for further understanding.

Proposed method

racefox

- **C4**:Let: $\sum_{YY|X_T}$denote CCO of (X_T, Y)in $(\tilde{H_1}, \tilde{k_1})$
  denote: $F_m = \tilde{H_1} + R = f + c : f \in \tilde{H_1}, c \in R$
  then: $Tr(\sum_{YY|X_T}) = \epsilon_{F_m}(X_T) = inf_{f \in F} E_{X,Y}(Y - f(X_T))^2$
- Given n samples $(x_1, y_1), (x_2, y_2), ...(x_n, y_n)$ the empirical estimate is given by:
  $Tr(\sum_{Y\tilde{Y}|X_T}^{(n)}) = trace\left\{ \sum_{Y\tilde{Y}}^{(n)} - \sum_{\tilde{Y},X_T}^{(n)} [\sum_{X_T^{\tilde{}},Y_T}^{(n)} + \epsilon I] \sum_{\tilde{X_T} Y}^{(n)} \right\}$
  $Tr(\sum_{Y\tilde{Y}|X_T}^{(n)}) = \epsilon_n trace\{G_Y[G_X + n\epsilon_n I_n]\}$
  where $G_X = (I_n - \frac{1}{n} I * I^T) K_{X_T} (I_n - \frac{1}{n} I * I^T)$
  and $G_Y = (I_n - \frac{1}{n} I * I^T) K_{Y_T} (I_n - \frac{1}{n} I * I^T)$
- WLG: $trace[G_Y(G_{X_T} + n\epsilon_n I_n)^{-1}] = trace[YY^T(G_{X_T} + n\epsilon_N I_N)^{-1}] = trace[Y^T(G_{X_T} + n\epsilon_N I_N)^{-1} Y]$
- Univariate Objective:$min_{|T|=m} \tilde{Q^{(n)}} = Y^T(G_{X_T} + n\epsilon_N I_N)^{-1} Y$
  where $Y = (y_1, y_2, ..., y_n)$ is a n-dimensional vector.

Not intended to be understood at a single slide. Check reference for further understanding.

Proposed method

racefox

- **C4**:Let: $\sum_{YY|X_T}$denote CCO of (X_T, Y)in $(\tilde{H_1}, \tilde{k_1})$
  denote: $F_m = \tilde{H_1} + R = f + c : f \in \tilde{H_1}, c \in R$
  then: $Tr(\sum_{YY|X_T}) = \epsilon_{F_m}(X_T) = inf_{f \in F} E_{X,Y}(Y - f(X_T))^2$
- Given n samples $(x_1, y_1), (x_2, y_2), ...(x_n, y_n)$ the empirical estimate is given by:
  $Tr(\sum_{\tilde{YY}|X_T}^{(n)}) = trace\left\{\sum_{\tilde{YY}}^{(n)} - \sum_{\tilde{Y},X_T}^{(n)}[\sum_{\tilde{X_T},Y_T}^{(n)} + \epsilon I]\sum_{\tilde{X_T}Y}^{(n)}\right\}$
  $Tr(\sum_{\tilde{YY}|X_T}^{(n)}) = \epsilon_n trace\{G_Y[G_X + n\epsilon_n I_n]\}$
  where $G_X = (I_n - \frac{1}{n} I * I^T)K_{X_T}(I_n - \frac{1}{n} I * I^T)$
  and $G_Y = (I_n - \frac{1}{n} I * I^T)K_{Y_T}(I_n - \frac{1}{n} I * I^T)$
- WLG: $trace[G_Y(G_{X_T} + n\epsilon_n I_n)^{-1}] = trace[YY^T(G_{X_T} + n\epsilon_N I_N)^{-1}] = trace[Y^T(G_{X_T} + n\epsilon_N I_N)^{-1}Y]$
- Univariate Objective:$min_{|T|=m}\tilde{Q^{(n)}} = Y^T(G_{X_T} + n\epsilon_N I_N)^{-1}Y$
  where $Y = (y_1, y_2, ..., y_n)$ is a n-dimensional vector.

Not intended to be understood at a single slide. Check reference for further understanding.

Proposed method

racefox

- **C4**:Let: $\sum_{YY|X_T}$denote CCO of (X_T, Y)in $(\tilde{H_1}, \tilde{k_1})$
  denote: $F_m = \tilde{H_1} + R = f + c : f \in \tilde{H_1}, c \in R$
  then: $Tr(\sum_{YY|X_T}) = \epsilon_{F_m}(X_T) = inf_{f \in F} E_{X,Y}(Y - f(X_T))^2$
- Given n samples $(x_1, y_1), (x_2, y_2), ...(x_n, y_n)$ the empirical estimate is given by:
  $Tr(\sum_{\tilde{YY}|X_T}^{(n)}) = trace\left\{\sum_{\tilde{YY}}^{(n)} - \sum_{\tilde{Y},X_T}^{(n)}[\sum_{\tilde{X_T},Y_T}^{(n)} + \epsilon I]\sum_{\tilde{X_T}Y}^{(n)}\right\}$
  $Tr(\sum_{\tilde{YY}|X_T}^{(n)}) = \epsilon_n trace\{G_Y[G_X + n\epsilon_n I_n]\}$
  where $G_X = (I_n - \frac{1}{n} I * I^T)K_{X_T}(I_n - \frac{1}{n} I * I^T)$
  and $G_Y = (I_n - \frac{1}{n} I * I^T)K_{Y_T}(I_n - \frac{1}{n} I * I^T)$
- WLG: $trace[G_Y(G_{X_T} + n\epsilon_n I_n)^{-1}] = trace[YY^T(G_{X_T} + n\epsilon_N I_N)^{-1}] = trace[Y^T(G_{X_T} + n\epsilon_N I_N)^{-1} Y]$
- Univariate Objective:$min_{|T|=m} \tilde{Q^{(n)}} = Y^T(G_{X_T} + n\epsilon_N I_N)^{-1} Y$
  where $Y = (y_1, y_2, ..., y_n)$ is a n-dimensional vector.

Not intended to be understood at a single slide. Check reference for further understanding.

Proposed method

- **C4**:Let: $\sum_{YY|X_T}$denote CCO of (X_T, Y)in $(\tilde{H_1}, \tilde{k_1})$
  denote: $F_m = \tilde{H_1} + R = f + c : f \in \tilde{H_1}, c \in R$
  then: $Tr(\sum_{YY|X_T}) = \epsilon_{F_m}(X_T) = inf_{f \in F} E_{X,Y}(Y - f(X_T))^2$
- Given n samples $(x_1, y_1), (x_2, y_2), ...(x_n, y_n)$ the empirical estimate is given by:
  $Tr(\sum_{\tilde{YY|X_T}}^{(n)}) = trace\left\{\sum_{\tilde{YY}}^{(n)} - \sum_{\tilde{Y,X_T}}^{(n)}[\sum_{\tilde{X_T,Y_T}}^{(n)} + \epsilon I]\sum_{\tilde{X_T Y}}^{(n)}\right\}$
  $Tr(\sum_{\tilde{YY|X_T}}^{(n)}) = \epsilon_n trace\{G_Y[G_X + n\epsilon_n I_n]\}$
  where $G_X = (I_n - \frac{1}{n} I * I^T) K_{X_T} (I_n - \frac{1}{n} I * I^T)$
  and $G_Y = (I_n - \frac{1}{n} I * I^T) K_{Y_T} (I_n - \frac{1}{n} I * I^T)$
- WLG: $trace[G_Y(G_{X_T} + n\epsilon_n I_n)^{-1}] = trace[YY^T(G_{X_T} + n\epsilon_N I_N)^{-1}] = trace[Y^T(G_{X_T} + n\epsilon_N I_N)^{-1} Y]$
- Univariate Objective:$min_{|T|=m} \tilde{Q^{(n)}} = Y^T(G_{X_T} + n\epsilon_N I_N)^{-1} Y$
  where $Y = (y_1, y_2, ..., y_n)$ is a n-dimensional vector.

Not intended to be understood at a single slide. Check reference for further understanding.

Proposed method

racefox

- **C4**:Let: $\sum_{YY|X_T}$denote CCO of (X_T, Y)in $(\tilde{H}_1, \tilde{k}_1)$
  denote: $F_m = \tilde{H}_1 + R = f + c : f \in \tilde{H}_1, c \in R$
  then: $Tr(\sum_{YY|X_T}) = \epsilon_{F_m}(X_T) = inf_{f \in F} E_{X,Y}(Y - f(X_T))^2$

- Given n samples $(x_1, y_1), (x_2, y_2), ...(x_n, y_n)$ the empirical estimate is given by:
  $Tr(\sum_{\tilde{YY}|X_T}^{(n)}) = trace\left\{\sum_{\tilde{YY}}^{(n)} - \sum_{\tilde{Y},X_T}^{(n)}[\sum_{\tilde{X_T},Y_T}^{(n)} + \epsilon I]\sum_{\tilde{X_T}Y}^{(n)}\right\}$
  $Tr(\sum_{\tilde{YY}|X_T}^{(n)}) = \epsilon_n trace\{G_Y[G_X + n\epsilon_n I_n]\}$
  where $G_X = (I_n - \frac{1}{n} I * I^T) K_{X_T} (I_n - \frac{1}{n} I * I^T)$
  and $G_Y = (I_n - \frac{1}{n} I * I^T) K_{Y_T} (I_n - \frac{1}{n} I * I^T)$

- WLG: $trace[G_Y(G_{X_T} + n\epsilon_n I_n)^{-1}] = trace[YY^T(G_{X_T} + n\epsilon_N I_N)^{-1}] = trace[Y^T(G_{X_T} + n\epsilon_N I_N)^{-1} Y]$

- Univariate Objective:$min_{|T|=m} \tilde{Q^{(n)}} = Y^T(G_{X_T} + n\epsilon_N I_N)^{-1} Y$
  where $Y = (y_1, y_2, ..., y_n)$ is a n-dimensional vector.

Not intended to be understood at a single slide. Check reference for further understanding.

Proposed method

racefox

- **C4**:Let: $\sum_{YY|X_T}$denote CCO of (X_T, Y)in $(\tilde{H_1}, \tilde{k_1})$
  denote: $F_m = \tilde{H_1} + R = f + c : f \in \tilde{H_1}, c \in R$
  then: $Tr(\sum_{YY|X_T}) = \epsilon_{F_m}(X_T) = inf_{f \in F} E_{X,Y}(Y - f(X_T))^2$

- Given n samples $(x_1, y_1), (x_2, y_2), ...(x_n, y_n)$ the empirical estimate is given by:
  $Tr(\sum_{\tilde{YY}|X_T}{}^{(n)}) = trace\left\{\sum_{\tilde{YY}}{}^{(n)} - \sum_{\tilde{Y},X_T}{}^{(n)}[\sum_{\tilde{X_T},Y_T}{}^{(n)} + \epsilon I]\sum_{\tilde{X_T}Y}{}^{(n)}\right\}$
  $Tr(\sum_{\tilde{YY}|X_T}{}^{(n)}) = \epsilon_n trace\{G_Y[G_X + n\epsilon_n I_n]\}$
  where $G_X = (I_n - \frac{1}{n}I * I^T)K_{X_T}(I_n - \frac{1}{n}I * I^T)$
  and $G_Y = (I_n - \frac{1}{n}I * I^T)K_{Y_T}(I_n - \frac{1}{n}I * I^T)$

- WLG: $trace[G_Y(G_{X_T} + n\epsilon_n I_n)^{-1}] = trace[YY^T(G_{X_T} + n\epsilon_N I_N)^{-1}] = trace[Y^T(G_{X_T} + n\epsilon_N I_N)^{-1}Y]$

- Univariate Objective:$min_{|T|=m}\tilde{Q^{(n)}} = Y^T(G_{X_T} + n\epsilon_N I_N)^{-1}Y$
  where $Y = (y_1, y_2, ..., y_n)$ is a n-dimensional vector.

Not intended to be understood at a single slide. Check reference for further understanding.

Proposed method

racefox

- **C4**:Let: $\sum_{YY|X_T}$denote CCO of (X_T, Y)in $(\tilde{H_1}, \tilde{k_1})$
  denote: $F_m = \tilde{H_1} + R = f + c : f \in \tilde{H_1}, c \in R$
  then: $Tr(\sum_{YY|X_T}) = \epsilon_{F_m}(X_T) = inf_{f \in F} E_{X,Y}(Y - f(X_T))^2$
- Given n samples $(x_1, y_1), (x_2, y_2), ...(x_n, y_n)$ the empirical estimate is given by:
  $Tr(\sum_{\tilde{YY}|X_T}^{(n)}) = trace\left\{\sum_{\tilde{YY}}^{(n)} - \sum_{\tilde{Y},X_T}^{(n)}[\sum_{X_{\tilde{T}},Y_T}^{(n)} + \epsilon I]\sum_{\tilde{X_T}Y}^{(n)}\right\}$
  $Tr(\sum_{\tilde{YY}|X_T}^{(n)}) = \epsilon_n trace\{G_Y[G_X + n\epsilon_n I_n]\}$
  where $G_X = (I_n - \frac{1}{n} I * I^T)K_{X_T}(I_n - \frac{1}{n} I * I^T)$
  and $G_Y = (I_n - \frac{1}{n} I * I^T)K_{Y_T}(I_n - \frac{1}{n} I * I^T)$
- WLG: $trace[G_Y(G_{X_T} + n\epsilon_n I_n)^{-1}] = trace[YY^T(G_{X_T} + n\epsilon_N I_N)^{-1}] = trace[Y^T(G_{X_T} + n\epsilon_N I_N)^{-1}Y]$
- Univariate Objective:$min_{|T|=m}\tilde{Q^{(n)}} = Y^T(G_{X_T} + n\epsilon_N I_N)^{-1}Y$
  where $Y = (y_1, y_2, ..., y_n)$ is a n-dimensional vector.

Not intended to be understood at a single slide. Check reference for further understanding.

Proposed method

racefox

- **C4**:Let: $\sum_{YY|X_T}$denote CCO of (X_T, Y)in $(\tilde{H_1}, \tilde{k_1})$
  denote: $F_m = \tilde{H_1} + R = f + c : f \in \tilde{H_1}, c \in R$
  then: $Tr(\sum_{YY|X_T}) = \epsilon_{F_m}(X_T) = inf_{f \in F} E_{X,Y}(Y - f(X_T))^2$
- Given n samples $(x_1, y_1), (x_2, y_2), ...(x_n, y_n)$ the empirical estimate is given by:
  $Tr(\sum_{\tilde{YY}|X_T}^{(n)}) = trace\left\{\sum_{\tilde{YY}}^{(n)} - \sum_{\tilde{Y},X_T}^{(n)}[\sum_{\tilde{X_T},Y_T}^{(n)} + \epsilon I]\sum_{\tilde{X_T}Y}^{(n)}\right\}$
  $Tr(\sum_{\tilde{YY}|X_T}^{(n)}) = \epsilon_n trace\{G_Y[G_X + n\epsilon_n I_n]\}$
  where $G_X = (I_n - \frac{1}{n} I * I^T) K_{X_T} (I_n - \frac{1}{n} I * I^T)$
  and $G_Y = (I_n - \frac{1}{n} I * I^T) K_{Y_T} (I_n - \frac{1}{n} I * I^T)$
- WLG: $trace[G_Y(G_{X_T} + n\epsilon_n I_n)^{-1}] = trace[YY^T(G_{X_T} + n\epsilon_N I_N)^{-1}] = trace[Y^T(G_{X_T} + n\epsilon_N I_N)^{-1} Y]$
- Univariate Objective:$min_{|T|=m} \tilde{Q^{(n)}} = Y^T(G_{X_T} + n\epsilon_N I_N)^{-1} Y$
  where $Y = (y_1, y_2, ..., y_n)$ is a n-dimensional vector.

Not intended to be understood at a single slide. Check reference for further understanding.

Proposed method

- **C4**:Let: $\sum_{YY|X_T}$denote CCO of (X_T, Y)in $(\tilde{H_1}, \tilde{k_1})$
  denote: $F_m = \tilde{H_1} + R = f + c : f \in \tilde{H_1}, c \in R$
  then: $Tr(\sum_{YY|X_T}) = \epsilon_{F_m}(X_T) = inf_{f \in F} E_{X,Y}(Y - f(X_T))^2$
- Given n samples $(x_1, y_1), (x_2, y_2), ...(x_n, y_n)$ the empirical estimate is given by:
  $Tr(\sum_{\tilde{YY}|X_T}^{(n)}) = trace\left\{\sum_{\tilde{YY}}^{(n)} - \sum_{\tilde{Y,X_T}}^{(n)}[\sum_{\tilde{X_T,Y_T}}^{(n)} + \epsilon I]\sum_{\tilde{X_T Y}}^{(n)}\right\}$
  $Tr(\sum_{\tilde{YY}|X_T}^{(n)}) = \epsilon_n trace\{G_Y[G_X + n\epsilon_n I_n]\}$
  where $G_X = (I_n - \frac{1}{n} I * I^T) K_{X_T} (I_n - \frac{1}{n} I * I^T)$
  and $G_Y = (I_n - \frac{1}{n} I * I^T) K_{Y_T} (I_n - \frac{1}{n} I * I^T)$
- WLG: $trace[G_Y(G_{X_T} + n\epsilon_n I_n)^{-1}] = trace[YY^T(G_{X_T} + n\epsilon_N I_N)^{-1}] = trace[Y^T(G_{X_T} + n\epsilon_N I_N)^{-1} Y]$
- Univariate Objective:$min_{|T|=m} \tilde{Q^{(n)}} = Y^T(G_{X_T} + n\epsilon_N I_N)^{-1} Y$
  where $Y = (y_1, y_2, ..., y_n)$ is a n-dimensional vector.

Not intended to be understood at a single slide. Check reference for further understanding.

Optimization

racefox

- $argmin_w : Y^T(G_{X_{w \odot X}} + n\epsilon_N I_N)^{-1} Y$ [7]
  subject to: $w_i \in \{0, 1\}, i = 1, .., d$
  where $I^T w \leq m$

[7] $\odot => Hadamard$

Optimization

racefox

- $argmin_w : Y^T(G_{X_{w \odot X}} + n\epsilon_N I_N)^{-1} Y$ [7]
  subject to: $w_i \in \{0, 1\}, i = 1, .., d$
  where $I^T w \leq m$

[7] $\odot =>$ *Hadamard*

Optimization

racefox

- $argmin_w : Y^T(G_{X_{w \odot X}} + n\epsilon_N I_N)^{-1} Y$ [7]
  subject to: $w_i \in \{0, 1\}, i = 1, .., d$
  where $I^T w \leq m$

[7] $\odot =>$ *Hadamard*

Optimization

- $argmin_w : Y^T (G_{X_{w \odot X}} + n\epsilon_N I_N)^{-1} Y$
  subject to : $w_i \in \{0, 1\}, i = 1, .., d$
  where $I^T w \leq m$

Optimization

racefox

- $argmin_w : Y^T(G_{X_{w \odot X}} + n\epsilon_N I_N)^{-1} Y$
  subject to $: w_i \in \{0, 1\}, i = 1, .., d$
  where $I^T w \leq m$

Optimization

racefox

- $argmin_w : Y^T(G_{X_{w \odot X}} + n\epsilon_N I_N)^{-1} Y$
  subject to $: w_i \in \{0, 1\}, i = 1, .., d$
  where $I^T w \leq m$

Optimization, w relaxation

racefox

- $argmin_w : Y^T (G_{X_{w \odot X}} + n\epsilon_N I_N)^{-1} Y$
  subject to $0 \leq w_i \leq 1, i = 1, .., d$
  where $I^T w \leq m$

Optimization,Removing the last inequality

racefox

- $argmin_w : Y^T(G_{X_{w \odot X}} + n\epsilon_N I_N)^{-1} Y + \lambda_1[I^T w - m]$
  subject to $0 \leq w_i \leq 1, i = 1, .., d$
  where $\lambda_1 \geq 0$

Optimization, Removing the matrix inversion

racefox

- $argmin_{w,\alpha} : \alpha * y + ||(G_{X_{w \odot X}} + n\epsilon_N I_N)\alpha + y||_2^2$
  subject to $0 \leq w_i \leq 1, i = 1, .., d$
  where $I^T w \leq m$ and $\alpha = (G_{X_{w \odot X}} + n\epsilon_N I_N)^{-1} y$

Optimization, kernel approximation

racefox

- $argmin_w$: $Y^T(G_{X_{w \odot x}} + n\epsilon_N I_N)^{-1}\ Y$
  subject to $0 \leq w_i \leq 1, i = 1, .., d$
  where $I^T w \leq m$
  $(G_{X_{w \odot x}} + n\epsilon_N I_N)^{-1} \approx \frac{1}{\epsilon_n n} I - \frac{1}{\epsilon_n^2 n^2} V(I_D + \frac{1}{\epsilon_n n} V_w^T V_w)^{-1} V_w$
  $(G_{X_{w \odot x}} + n\epsilon_N I_N)^{-1} \approx \frac{1}{\epsilon_n n}(I - V_w(V_w^T V_w + \epsilon_n b I_D)^{-1} V_w^T)$

Optimization, kernel approximation

racefox

- $argmin_w : Y^T(G_{X_{w\odot x}} + n\epsilon_N I_N)^{-1}\ Y$
  subject to $0 \leq w_i \leq 1, i = 1, .., d$
  where $I^T w \leq m$
  $(G_{X_{w\odot x}} + n\epsilon_N I_N)^{-1} \approx \frac{1}{\epsilon_n n} I - \frac{1}{\epsilon_n^2 n^2} V(I_D + \frac{1}{\epsilon_n n} V_w^T V_w)^{-1} V_w$
  $(G_{X_{w\odot x}} + n\epsilon_N I_N)^{-1} \approx \frac{1}{\epsilon_n n}(I - V_w(V_w^T V_w + \epsilon_n b I_D)^{-1} V_w^T)$

Optimization, kernel approximation

racefox

- $argmin_w : Y^T(G_{X_{w\odot X}} + n\epsilon_N I_N)^{-1}\ Y$
  subject to $0 \leq w_i \leq 1, i = 1, .., d$
  where $I^T w \leq m$
  $(G_{X_{w\odot X}} + n\epsilon_N I_N)^{-1} \approx \frac{1}{\epsilon_n n} I - \frac{1}{\epsilon_n^2 n^2} V(I_D + \frac{1}{\epsilon_n n} V_w^T V_w)^{-1} V_w$
  $(G_{X_{w\odot X}} + n\epsilon_N I_N)^{-1} \approx \frac{1}{\epsilon_n n}(I - V_w(V_w^T V_w + \epsilon_n b I_D)^{-1} V_w^T)$

racefox

Longer. Faster. Forever.


Thank
You
Merci
Gracias
Danke
Spacibo
Arigatô
Hvala
KamSahHamnida
TerimaKasih
Takk
aDank
Mèsi
Xièxie
Grazie
Chokrane
Kiitos
Dziakuju
BarakAllahuFiik
Rahmat
ThintKo
Buznyg
Blagodaram
Matondo
Mercé
Taiku
Dakujem
Gràcies
Faleminderit
Efharisto
NajisTuke
Grandmercé
Tanemirt
Tenki
Obrigado
Bayarlalaa
Nirringrazzjak
Salamat
Saha
Multumesc
CámOn
Gracie
GratiasAgo
GratiasAgimus
TänanVäga
Tak
Dankie
Sagolun
Motashakkeram
KopKhunKha
Mauruuru
Chnorakaloutioun
Gratzias
Barkal
Aabhar
Aćiū
Murakoze
Köszönöm
TapadhLeat
WaadMahadsantahay
Grazzi
Dhanyavaad
Mamnun
aNiKié