Inherently interpretable models

Szymon Bobek

Jagiellonian University
2024

JAGIELLONIAN UNIVERSITY
IN KRAKÓW

https://geist.re

Clever Hans

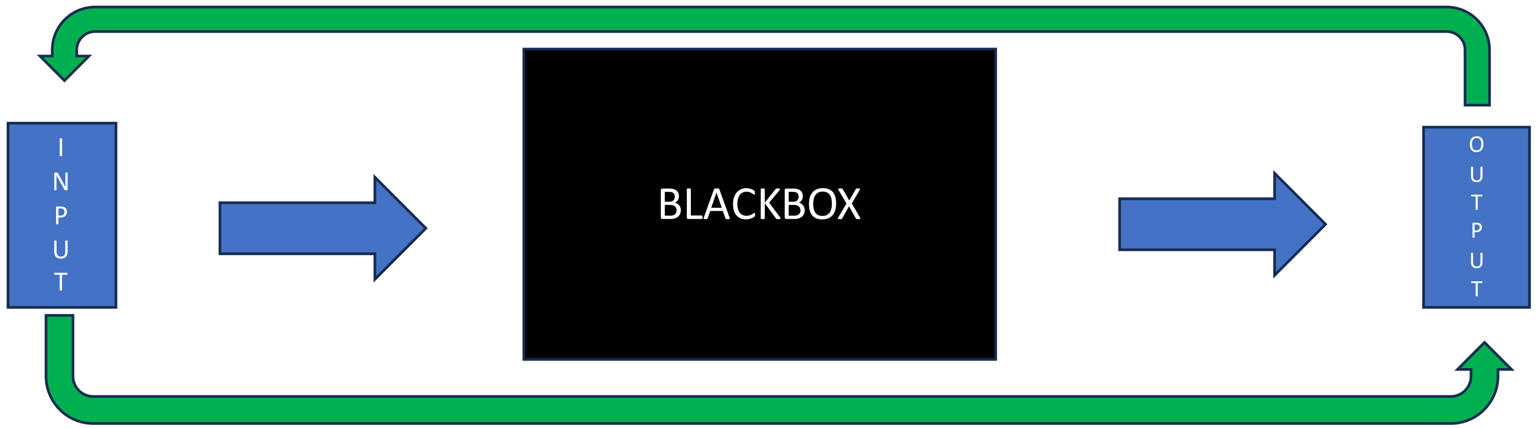
- Clever Hans, a horse, amazed audiences with apparent intelligence in early 1900s Germany.
- Claimed to solve math problems and answer questions.
- Drew significant public attention and curiosity about abilities.
- Attracted interest from scientists and psychologists studying cognition.
- Ultimately revealed reliance on subtle human cues, not intelligence.

"If the eighth day of the month comes on a Tuesday, what is the date of the following Friday?" Hans would answer by tapping his hoof eleven times.

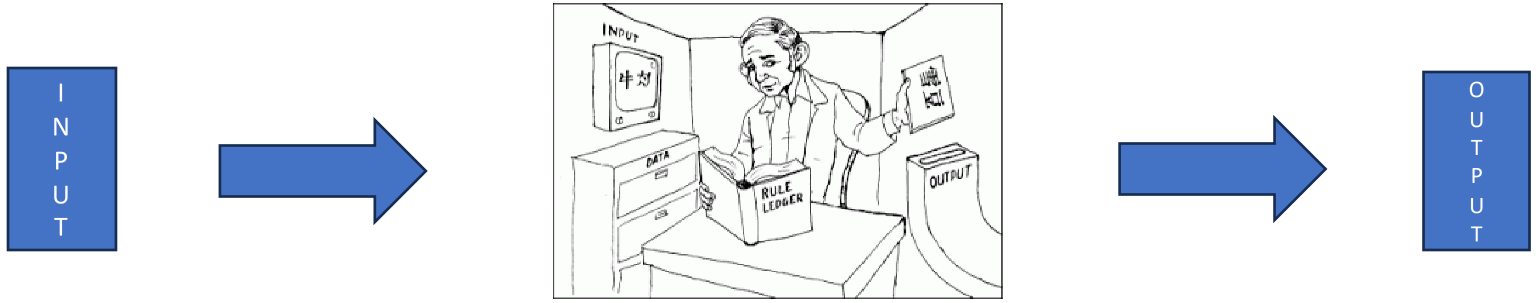
Glassbox and Blackbox

Glassbox vs Blackbox

XAI

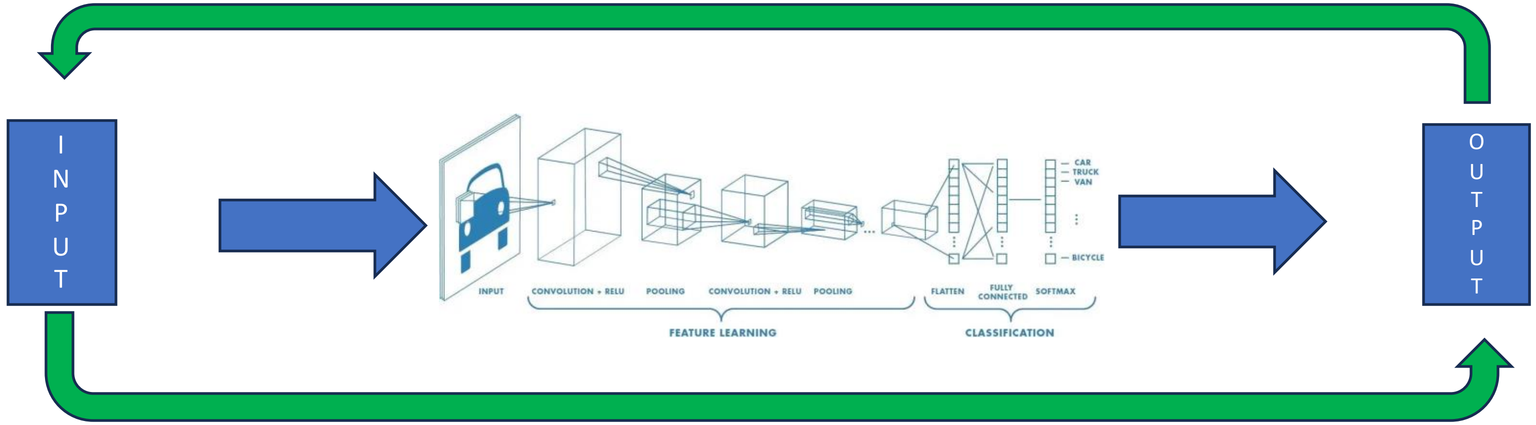


Glassbox

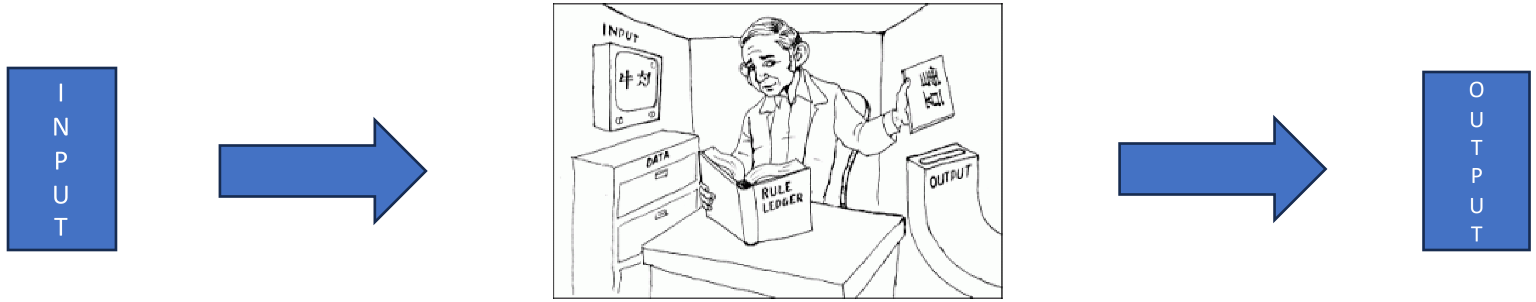


Glassbox vs Blackbox

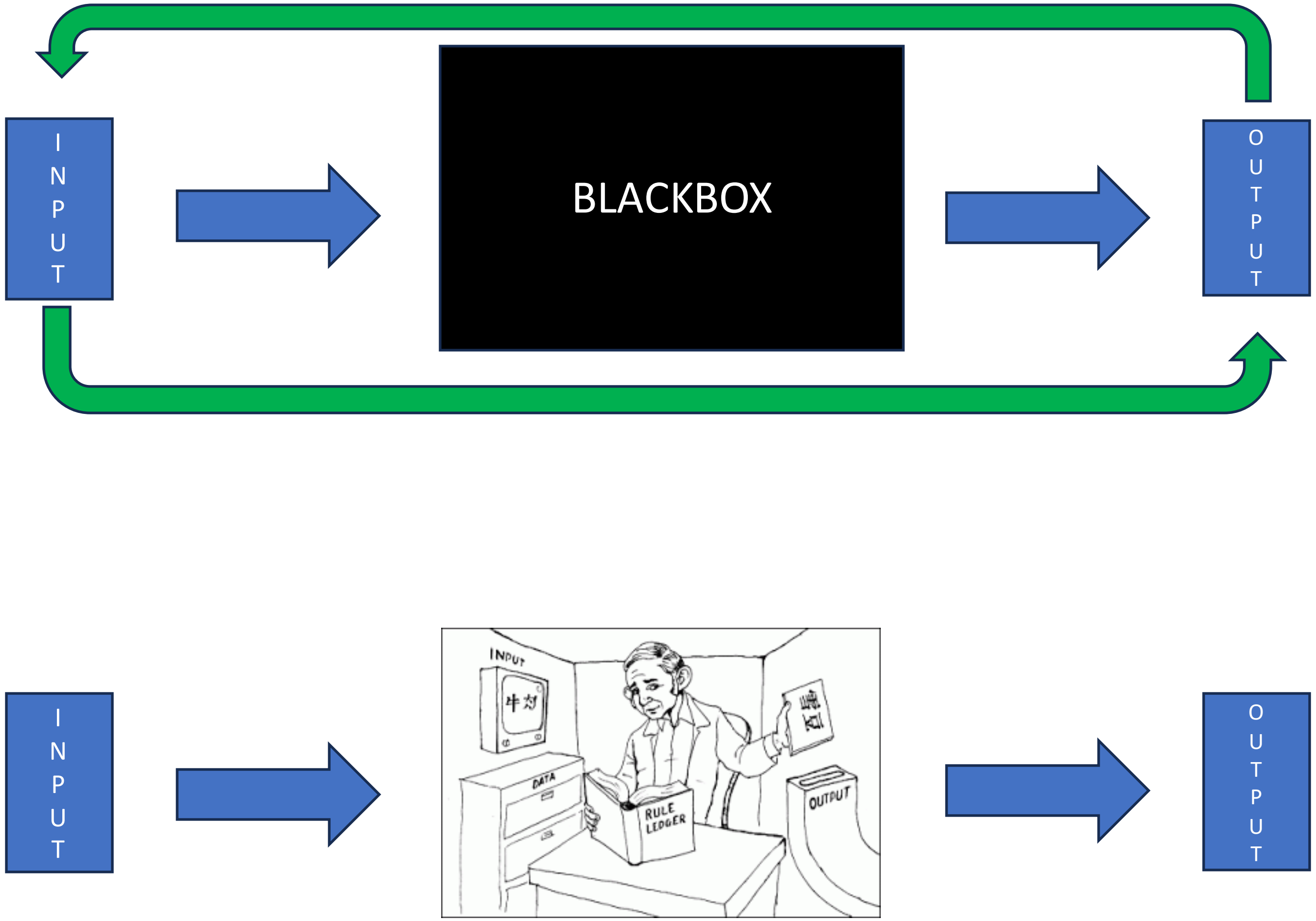
XAI



Glassbox



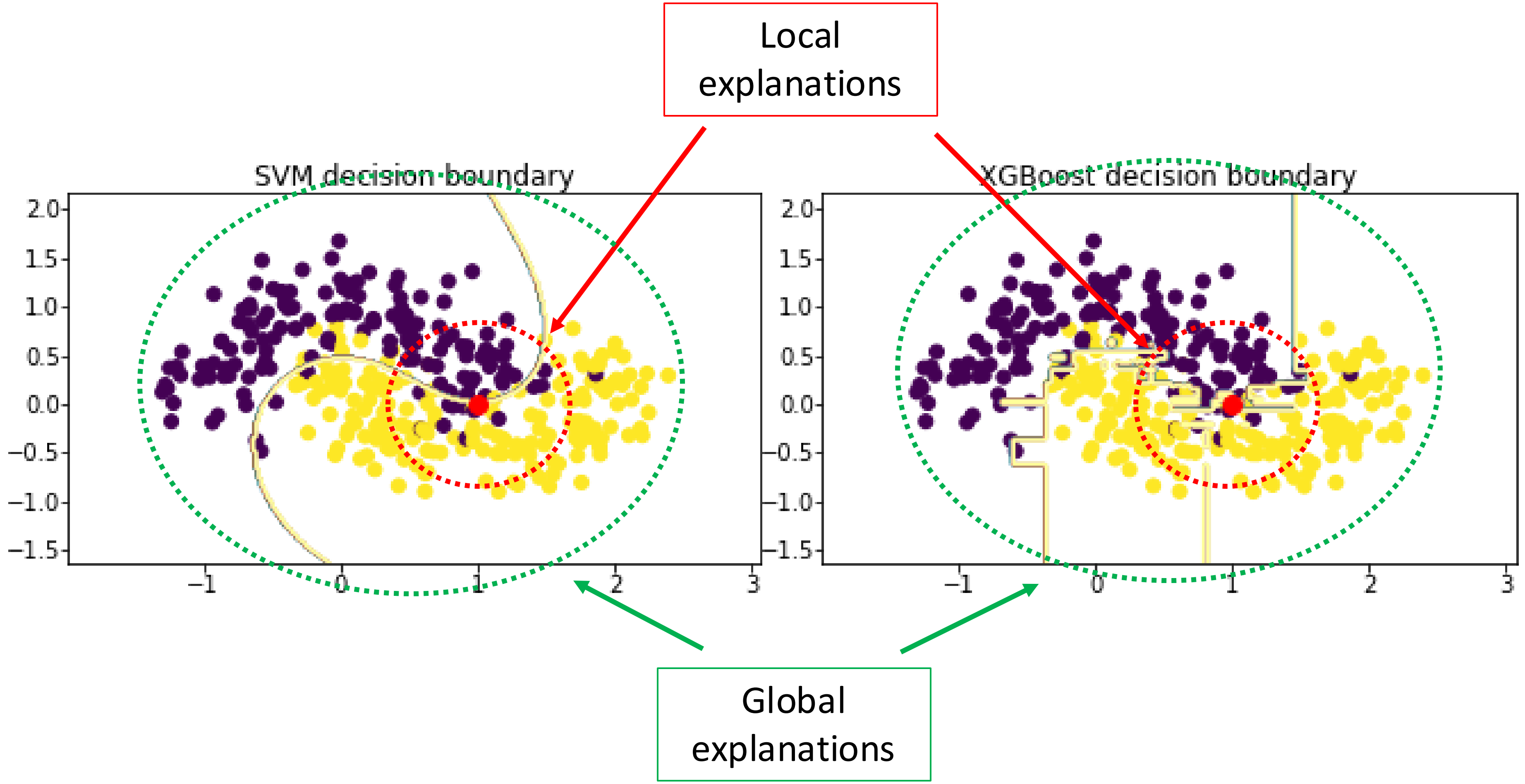
Glassbox vs Blackbox
Intrerpretability vs Explainability



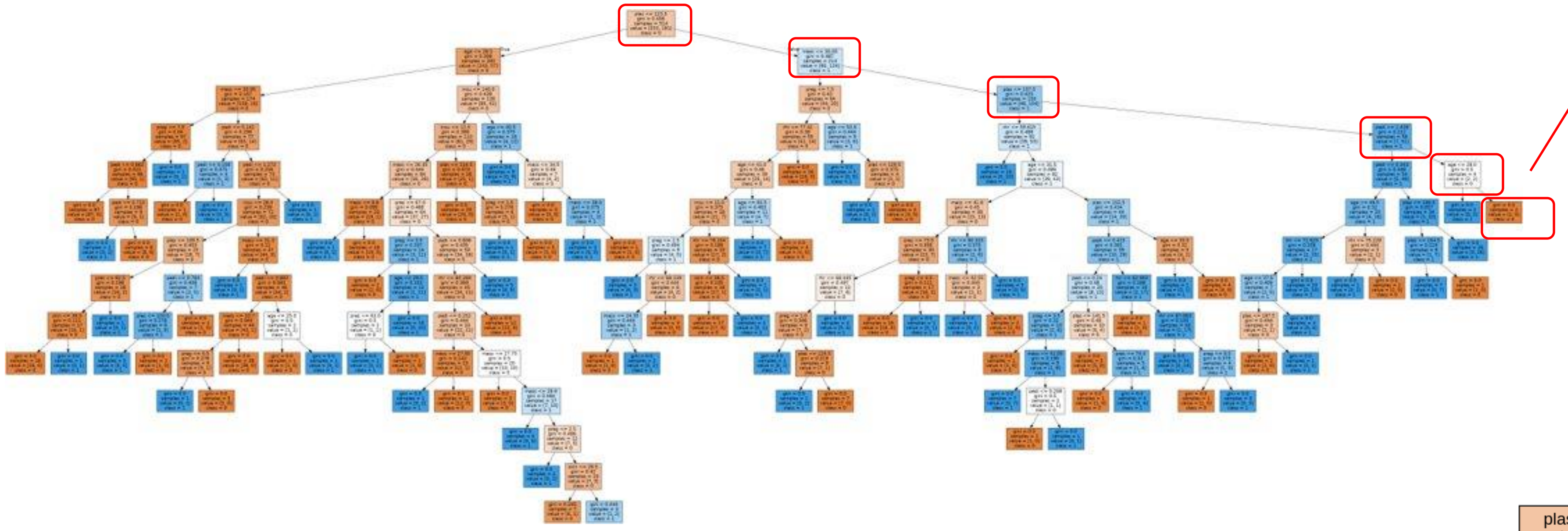
Interpretability
- It is a property of a model
- Interpretable models can provide explainability

Explainability
- It is a property of a decission process
- Explainability can be achieved even for not interpretable models

Local vs Global explanations



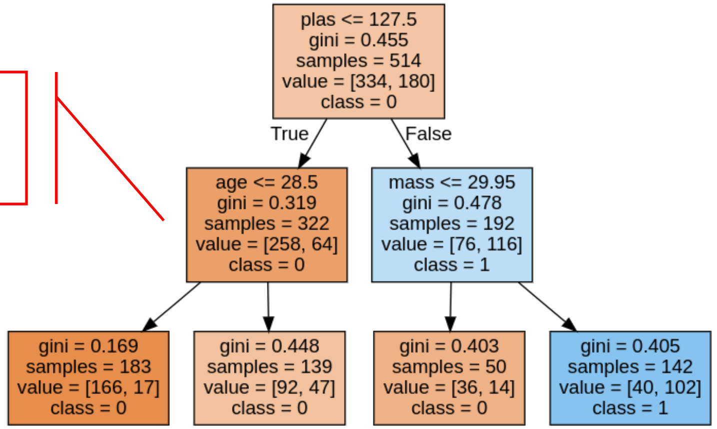
Locally and globally interpretable models



For a single instance it might be possible to interpret the model, but globally it might be difficult to grasp the model behaviour easily.

- Interpretability does not guarantee understandability!
- It depends on many factors

The model is simple enough to interpret it globally



Linear regression

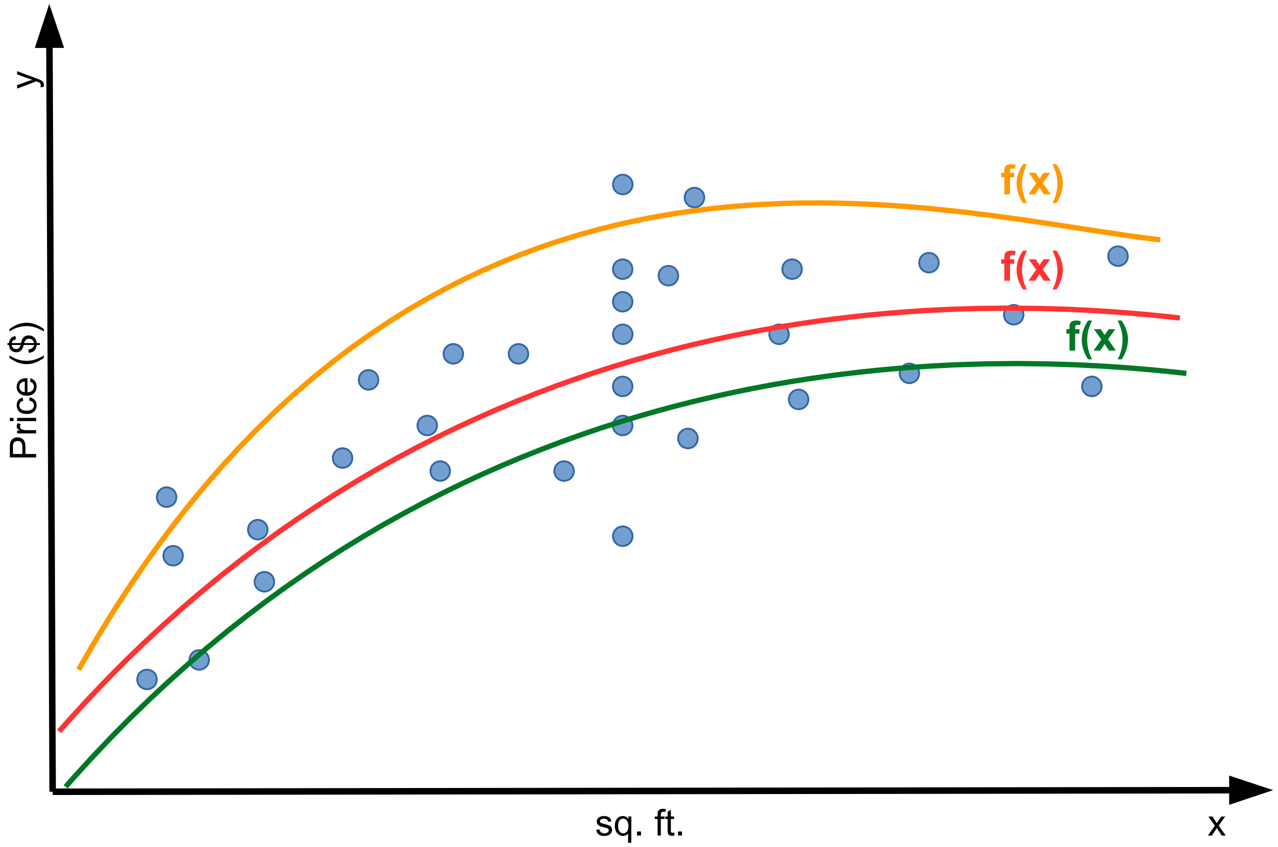
Linear regression

X_1 = 150 ft^2, y_1 = 300 000 $

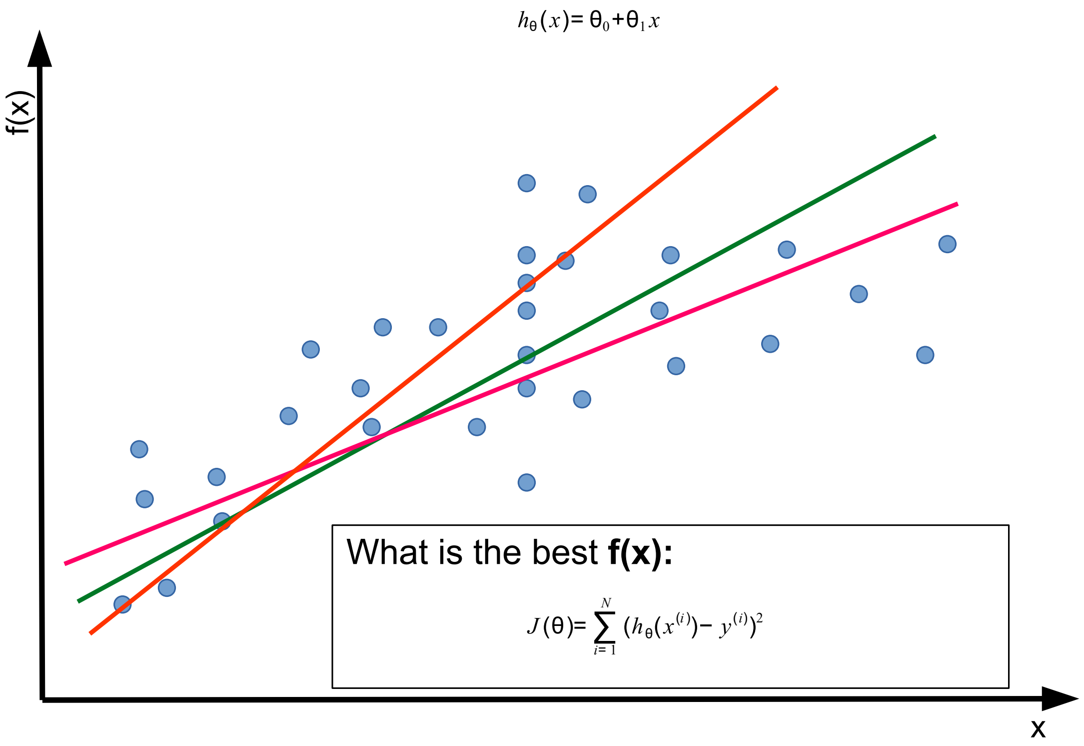
X_2 = 150 ft^2, y_2 = 150000 $

X_3 = 300 ft^2, y_3 = 1 000 000 $

X_4 = 170 ft^2, y_4 = 270 000 $

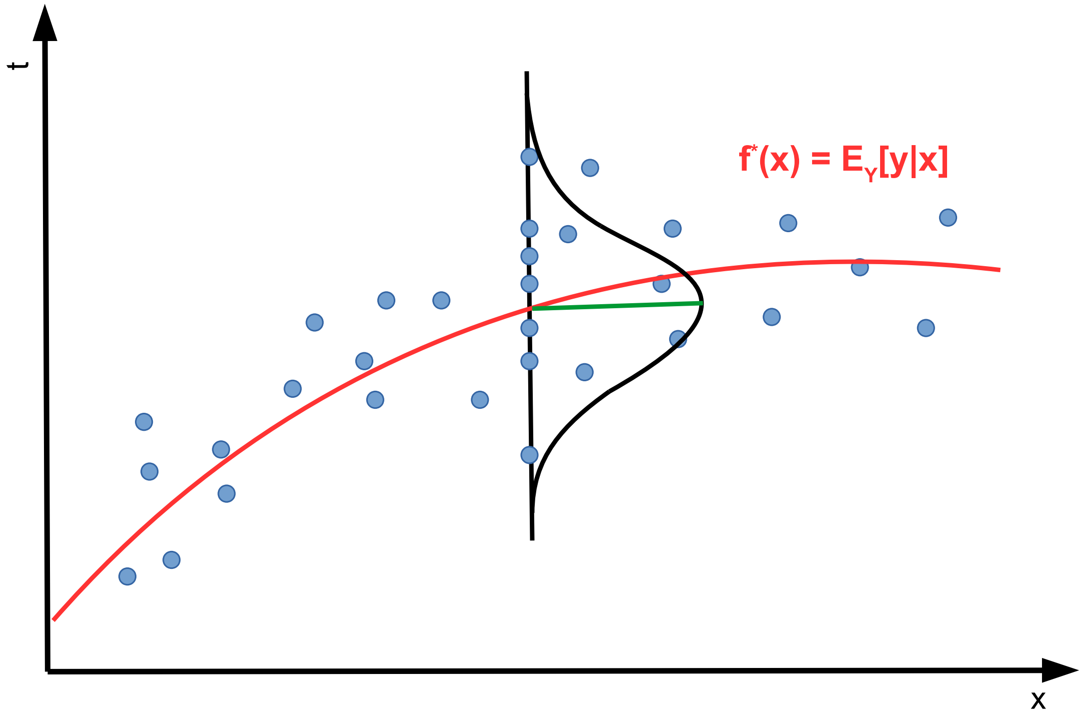


Linear regression



- Assumptions:
  - Linearity – interactions and nonlinearities need to be engineered
  - Normality - outcome, given features follows normal distribution
  - Homoscedasticity (constant variance) - the classic i.i.d assumption
  - Independence – the classic i.i.d assumption
  - Fixed features – no measurement errors assumed
  - Absence of multicolinearity – correlated features break the interpretability

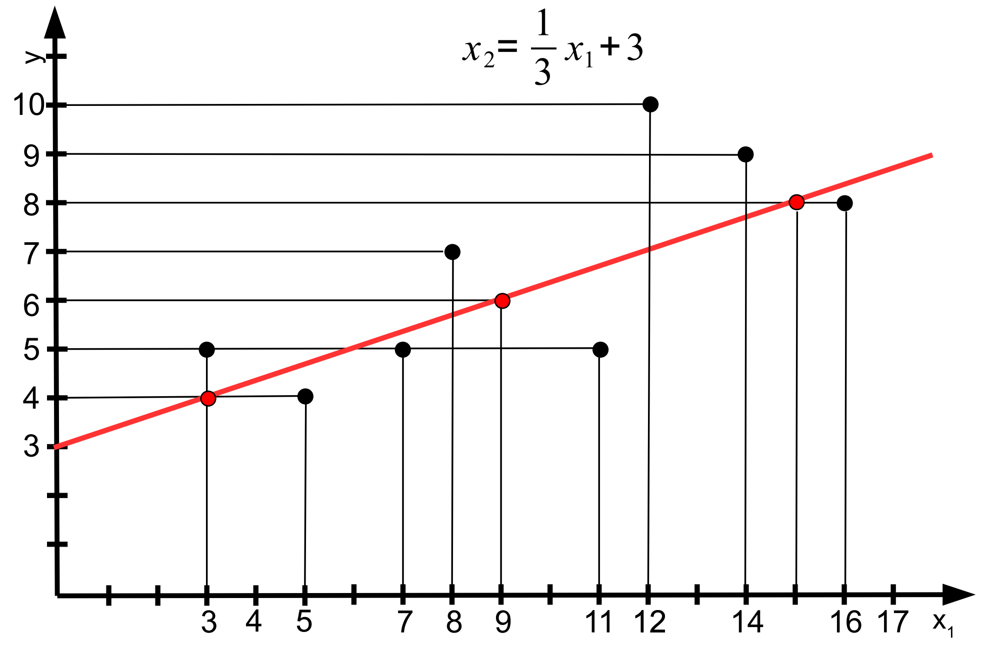
Linear regression



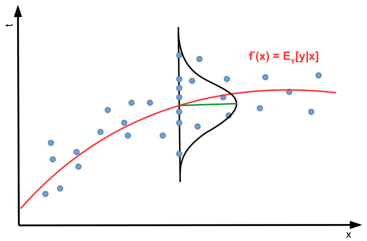
- Assumptions:
  - Linearity – interactions and nonlinearities need to be engineered
  - Normality - outcome, given features follows normal distribution
  - Homoscedasticity (constant variance) - the classic i.i.d assumption
  - Independence – the classic i.i.d assumption
  - Fixed features – no measurement errors assumed
  - Absence of multicolinearity – correlated features break the interpretability

How to interpret linear regression

- Interpretation of numerical features
- Interpretation of categorical features
- Feature importance
- **"All other features remain the same"**



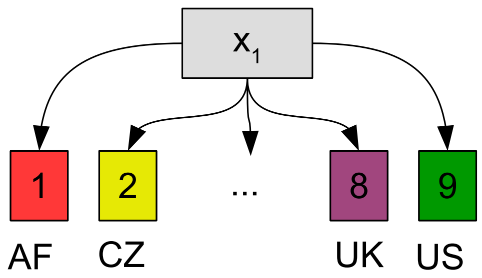
$$y = X\theta + \epsilon$$

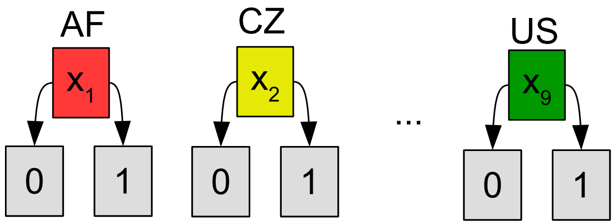


$$\hat{\theta} = (X^T X)^{-1} X^T y$$

$$SE(\hat{\theta}) = \sqrt{\hat{\sigma}^2 (X^T X)^{-1}}$$

$$t_{\hat{\theta}_j} = \frac{\hat{\theta}_j}{SE(\hat{\theta}_j)}$$

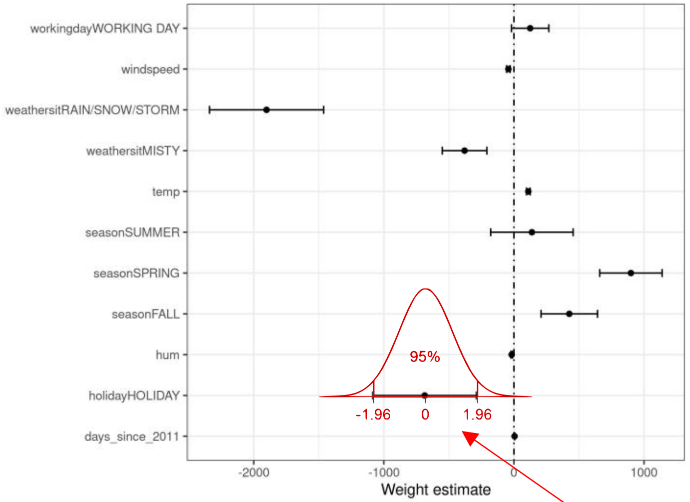




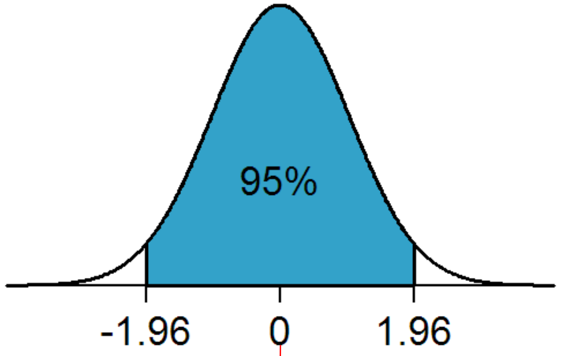
We do not know the real error (noise), so we use MSE as an estimate. Nice video explaining this: Video.
Note: You need to get diagonal values of $(X^T X)^{-1}$, as this is covariance matrix.

How to interpret linear regression

- Confidence intervals
- Effect plots
- Explain single instance

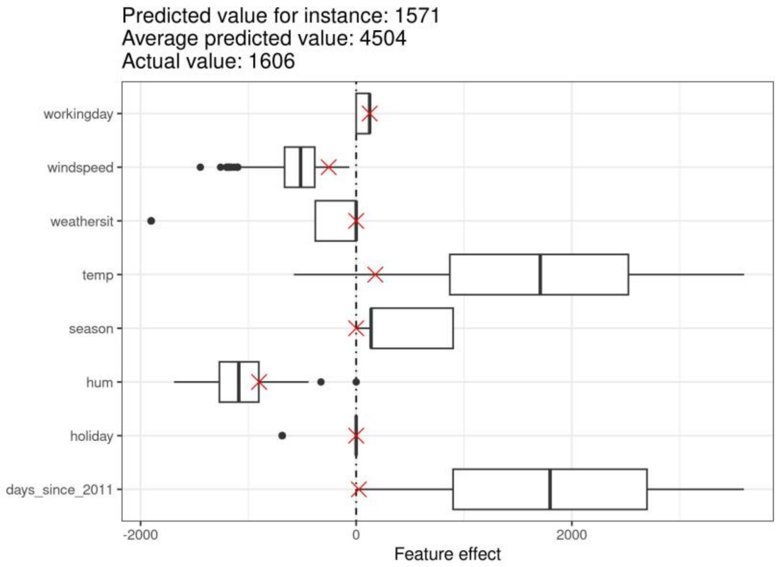


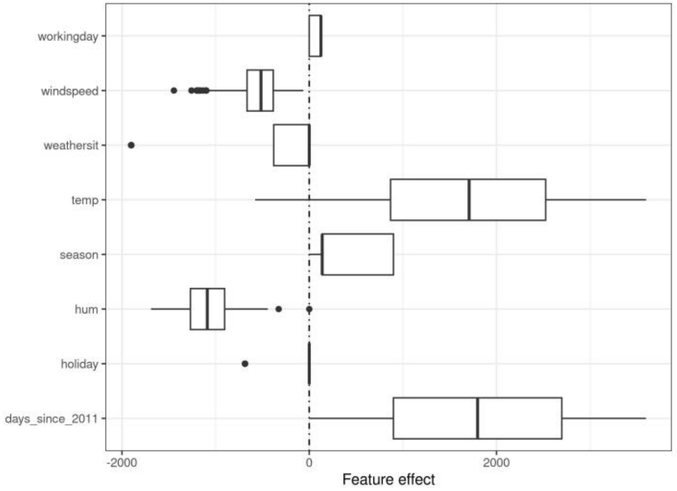
$$SE(\hat{\theta}) = \sqrt{\hat{\sigma}^2 (X^T X)^{-1}}$$



$$\hat{\theta} = (X^T X)^{-1} X^T y$$

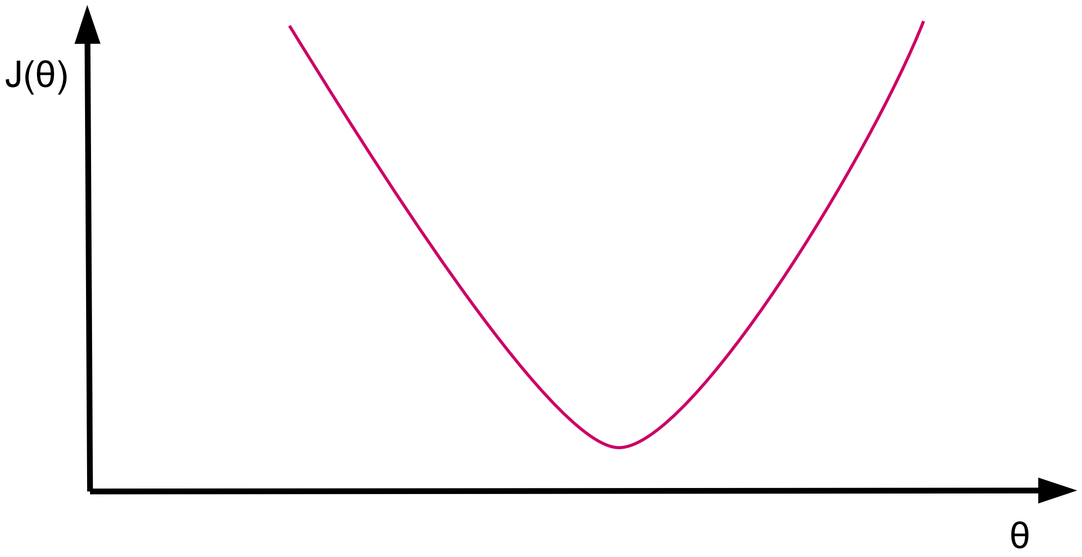
$$\hat{\theta}_i \pm 1.96 \cdot SE(\hat{\theta}_j)$$





Interpretability issues

- **OLS will give different results than gradient methods, because of normalization issues**
- Multicolinearity can break the interpretability
- Model is not human-interpretable when interactions and transformations are added



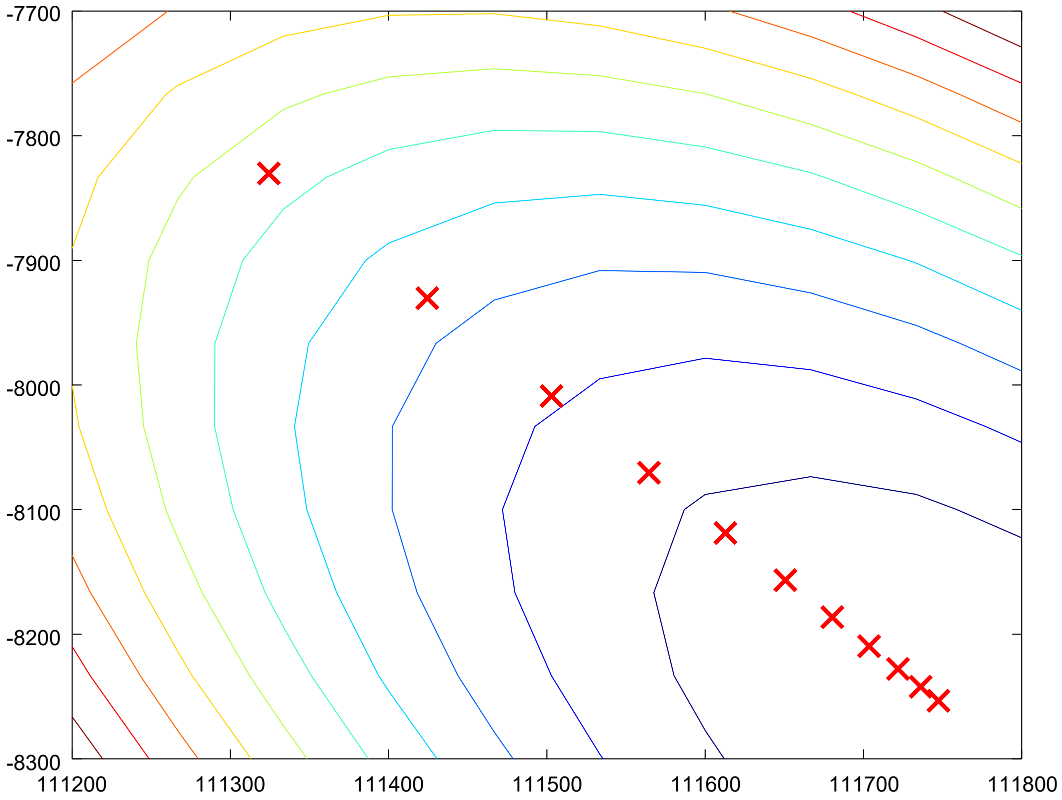
$$J(\theta) = \frac{1}{N}\sum_{i}^{N}(y^{(i)} - \theta x^{(i)})^2$$

$$\frac{\partial J(\theta)}{\partial \theta_i} = 2\sum_{j}^{N}(\theta x^{(j)} - y^{(j)})x_i^{(j)}$$

$$\theta_i = \theta_i - \alpha\frac{\partial J(\theta)}{\partial \theta_i}$$

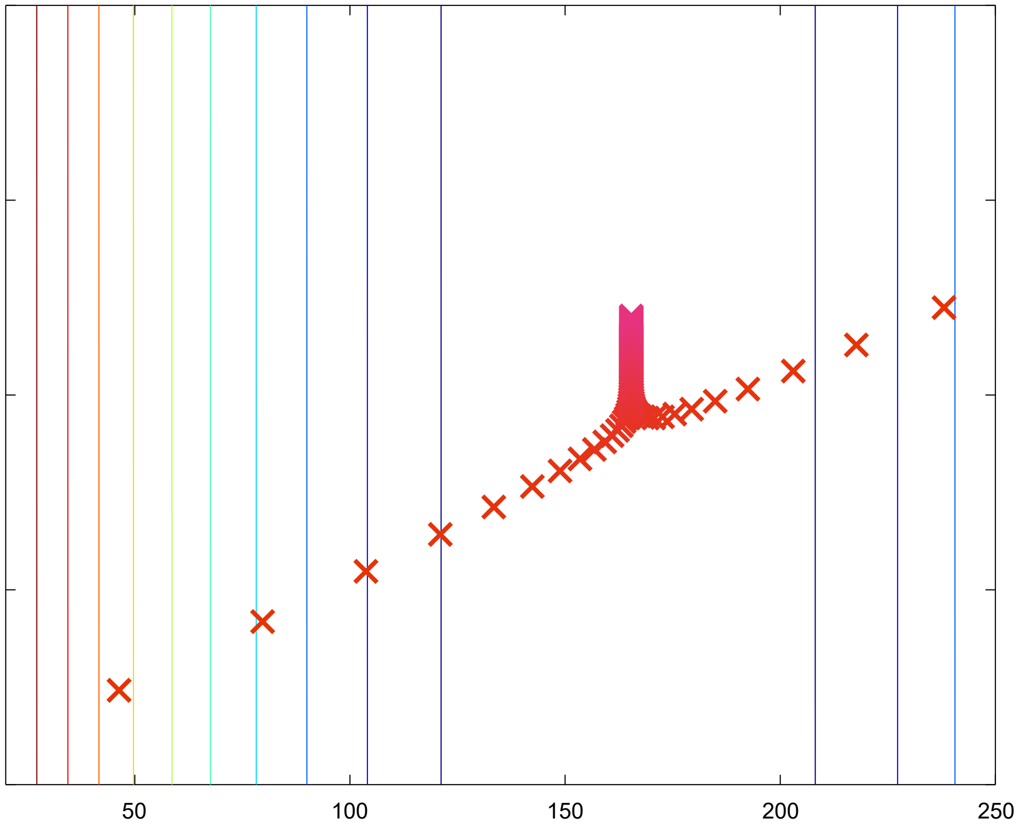
Interpretability issues

- **OLS will give different results than gradient methods, because of normalization issues**
- Multicolinearity can break the interpretability
- Model is not human-interpretable when interactions and transformations are added



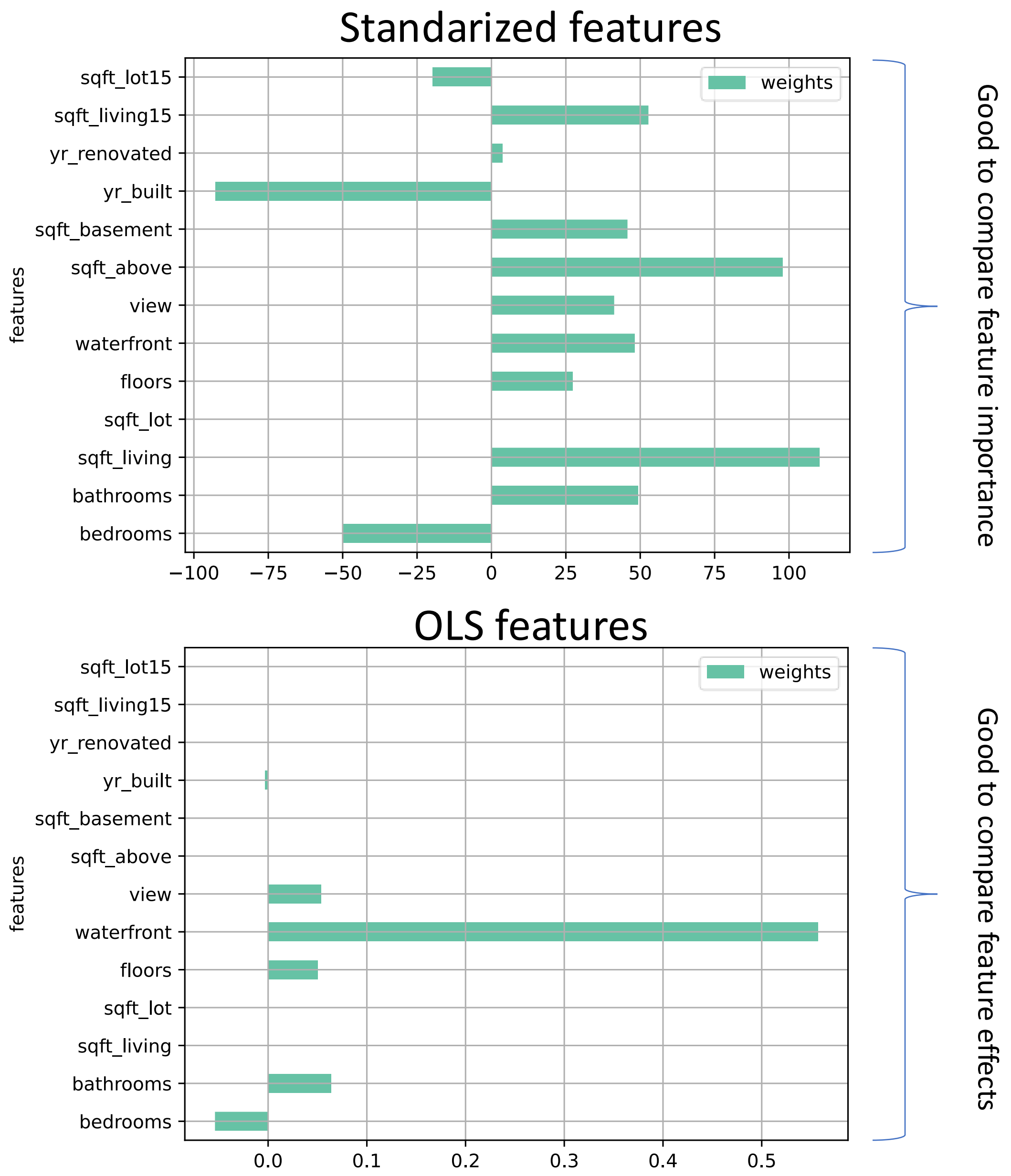
Interpretability issues

- **OLS will give different results than gradient methods, because of normalization issues**
- Multicolinearity can break the interpretability
- Model is not human-interpretable when interactions and transformations are added



Interpretability issues

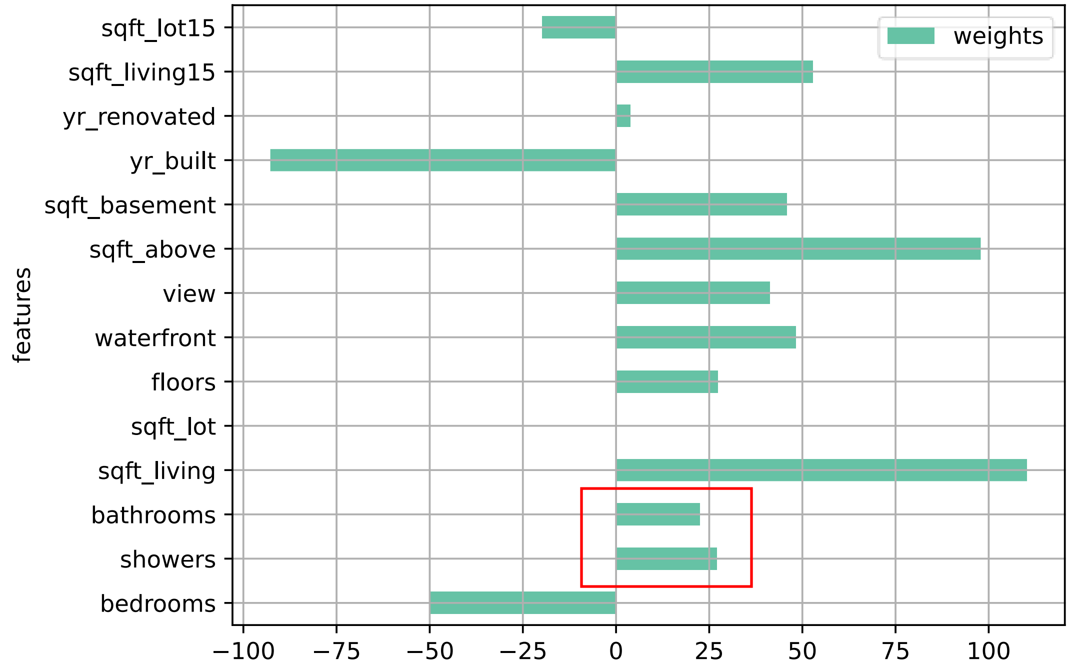
- **OLS will give different results than gradient methods, because of normalization issues**
- Multicolinearity can break the interpretability
- Model is not human-interpretable when interactions and transformations are added



The coefficients of the linear regression model (let's denote them as β_j) represent the expected change in the target variable Y for a one-standard-deviation increase in the predictor variable Z_j, holding all other variables constant.

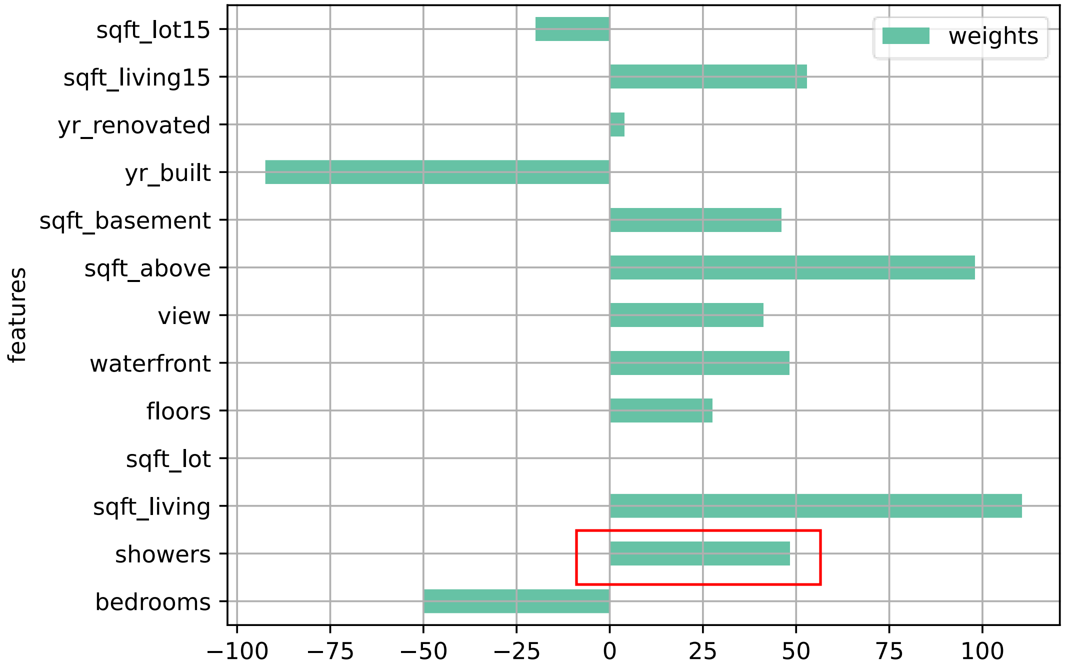
Interpretability issues

- OLS will give different results than gradient methods, because of normalization issues
- **Multicolinearity can break the interpretability**
- Model is not human-interpretable when interactions and transformations are added



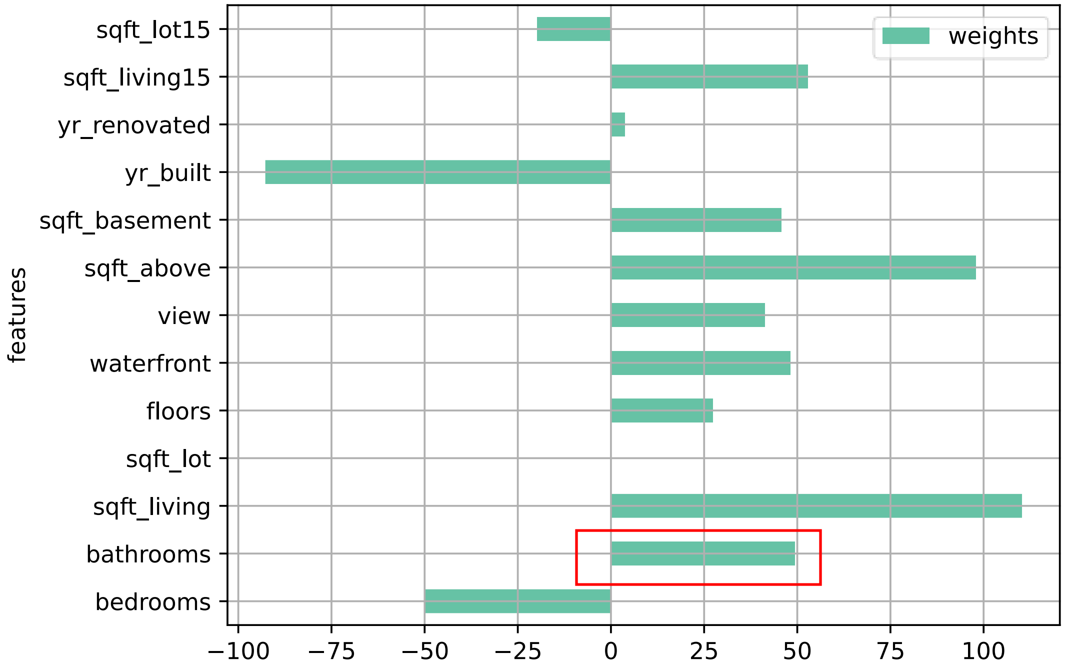
Interpretability issues

- OLS will give different results than gradient methods, because of normalization issues
- **Multicolinearity can break the interpretability**
- Model is not human-interpretable when interactions and transformations are added



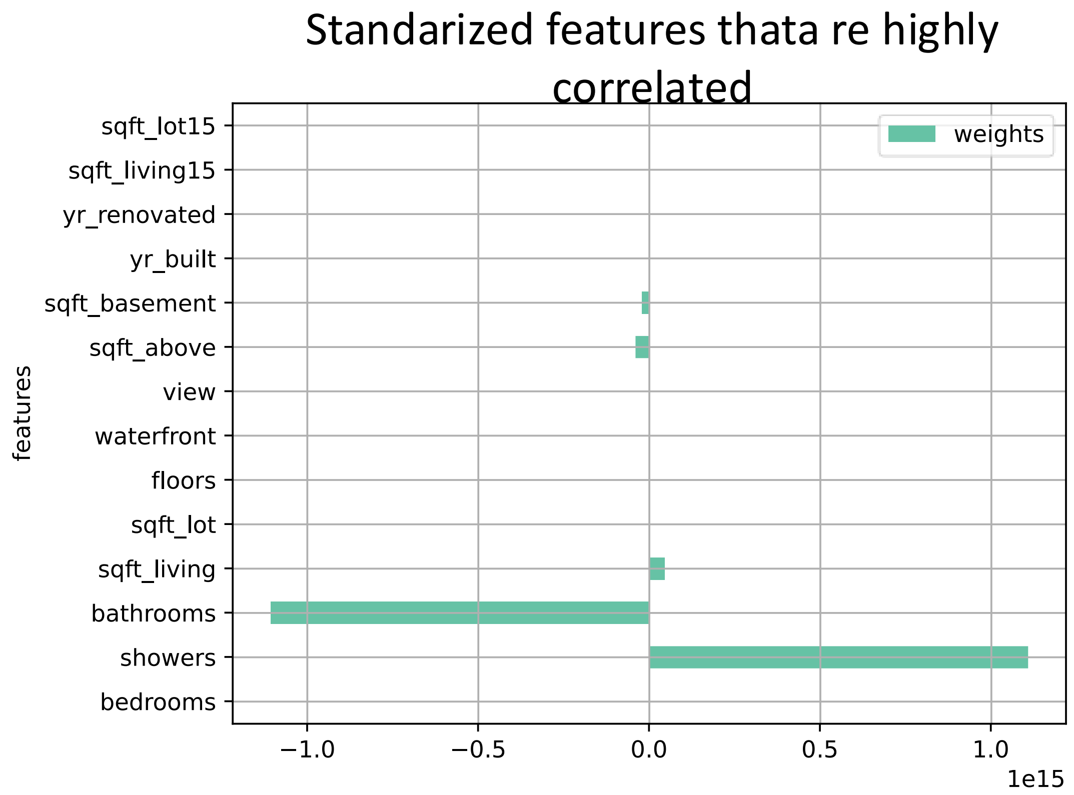
Interpretability issues

- OLS will give different results than gradient methods, because of normalization issues
- **Multicolinearity can break the interpretability**
- Model is not human-interpretable when interactions and transformations are added



Interpretability issues

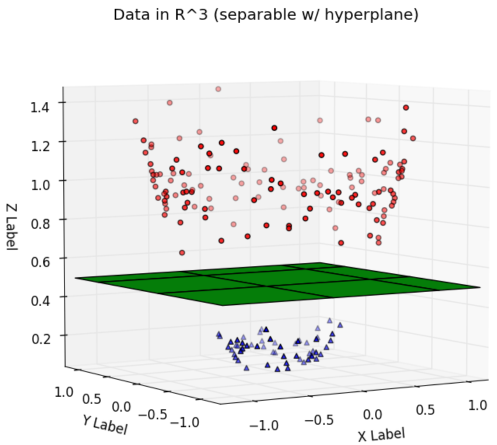
- OLS will give different results than gradient methods, because of normalization issues
- **Multicolinearity can break the interpretability**
- Model is not human-interpretable when interactions and transformations are added

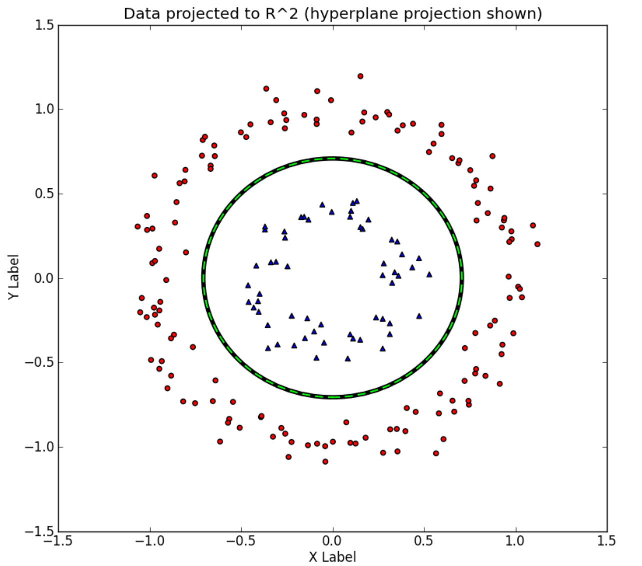


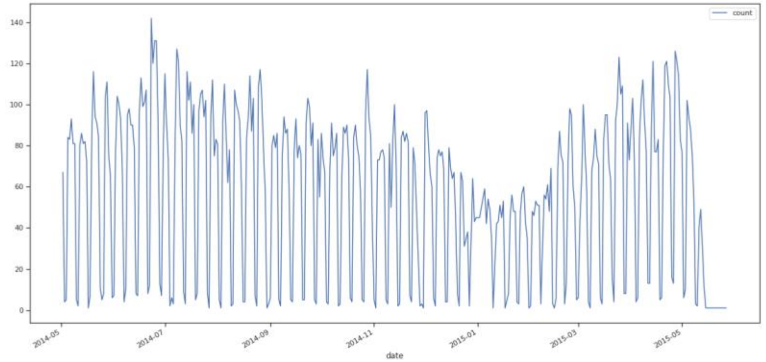
The model may compensate for this redundancy by inflating the coefficients of the correlated features to capture the shared variance. Consequently, the weights can appear significantly higher than they would for less correlated features.

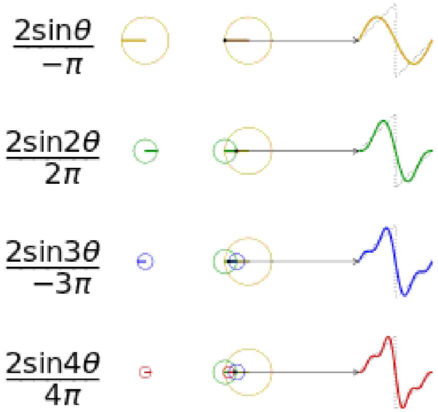
Interpretability issues

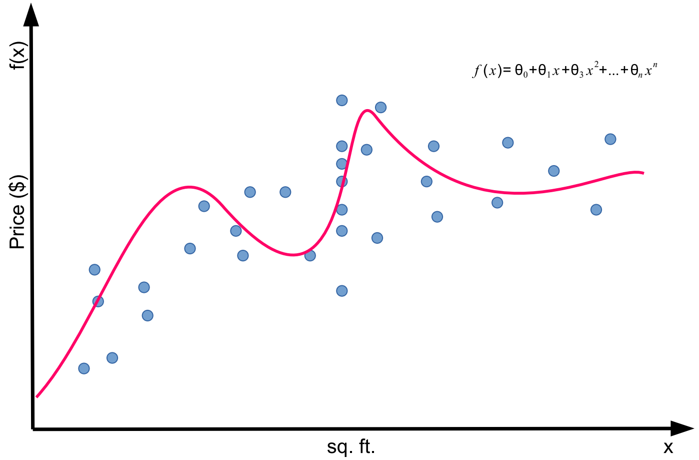
- OLS will give different results than not gradient methods, because of normalization issues
- Multicolinearity can break the interpretability
- **Model is not human-interpretable when interactions and transformsations are added**





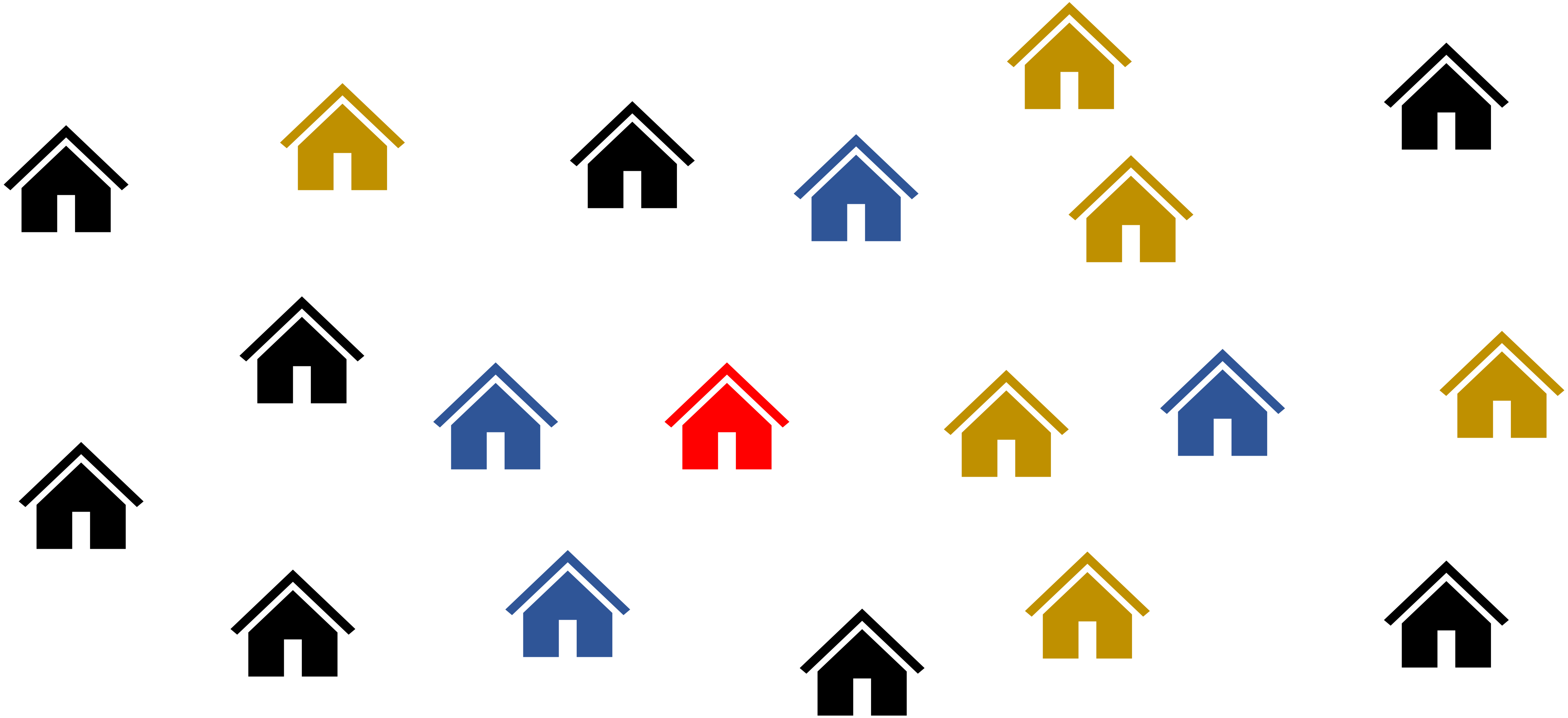




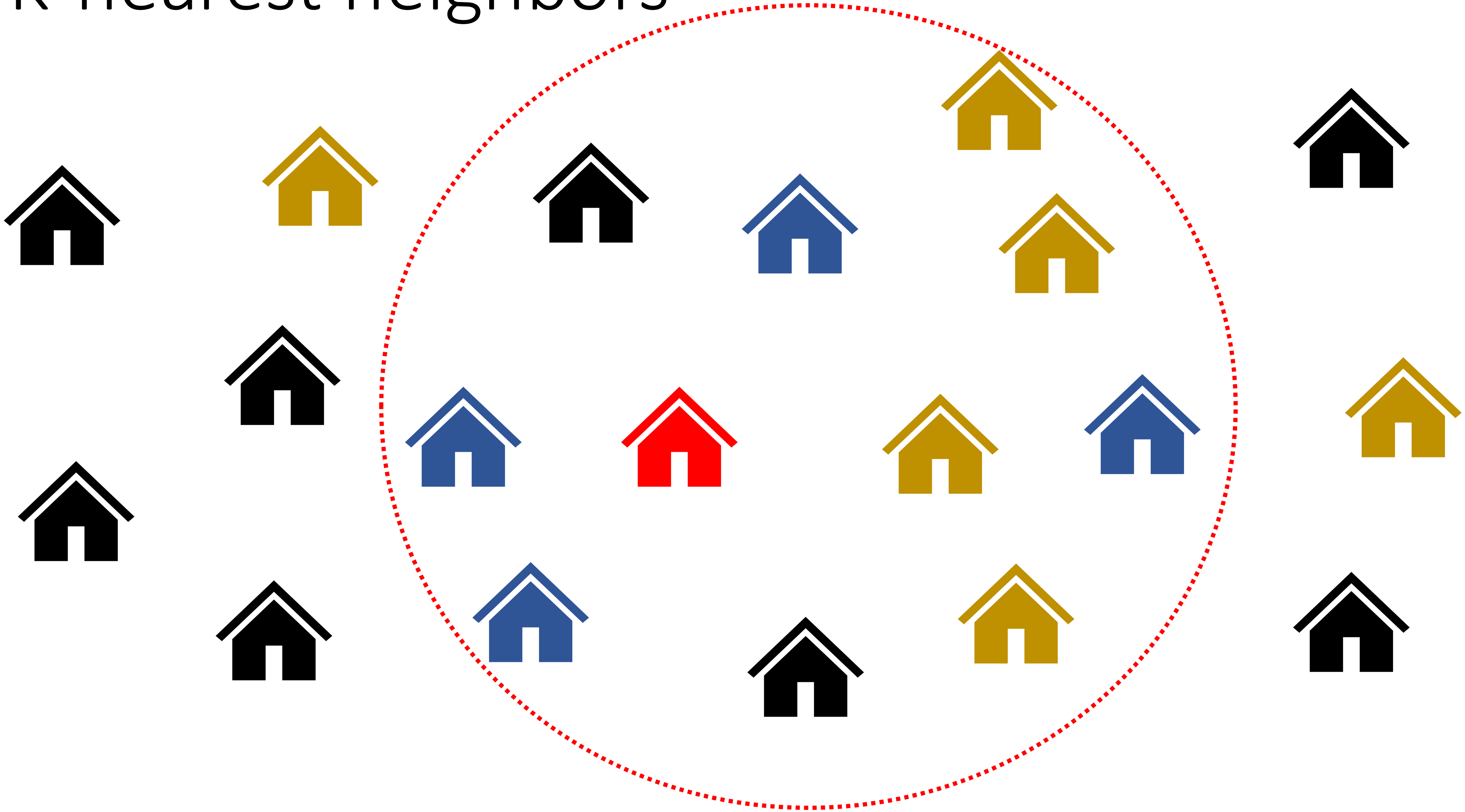


K-nearest neighbors

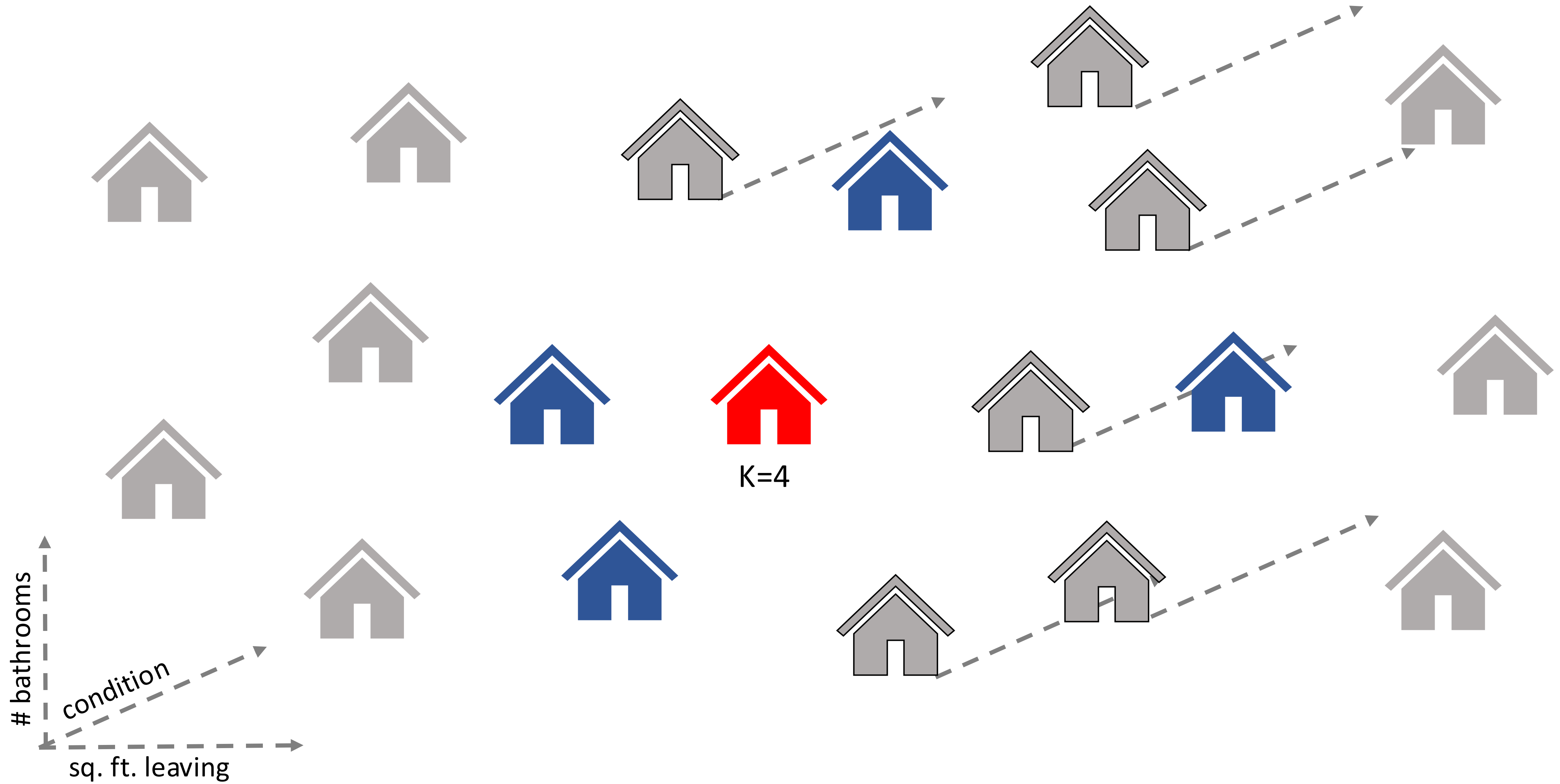
K-nearest neighbors



K-nearest neighbors

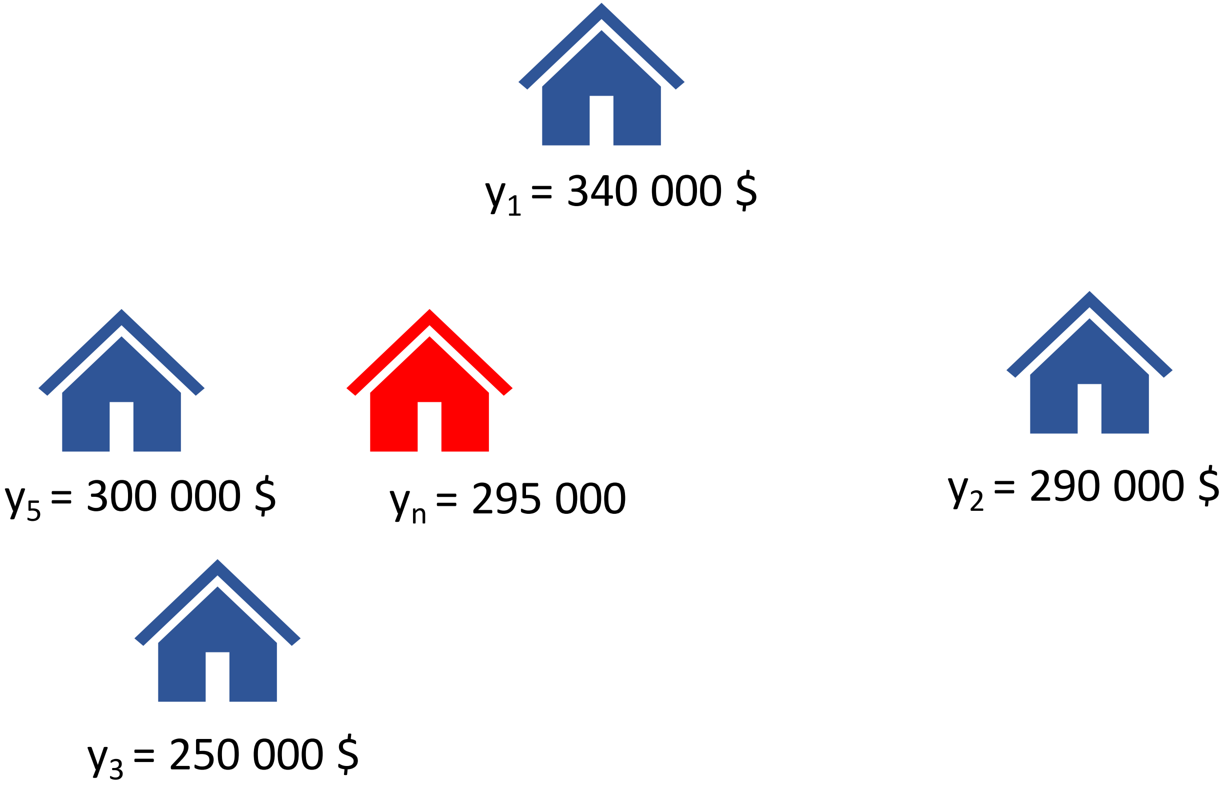


K-nearest neighbors



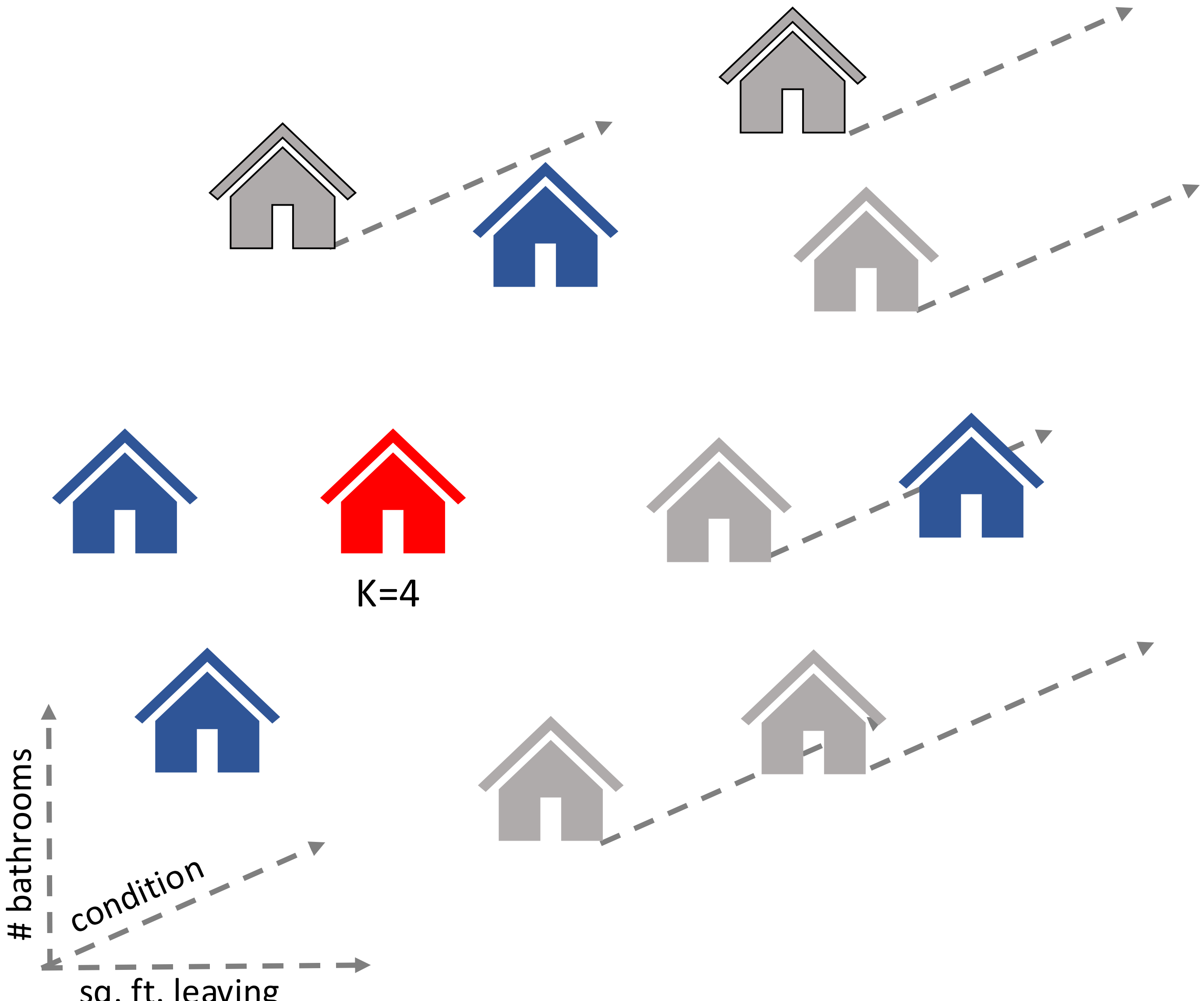
K-nearest neighbors

- Exaplain by example: the price of the house was estimated to 295 000 $ because most similar houses had prices from a range 250 000$ to 340 000 $
- Explain by explicitly providing K nearest neighbours for analysis



K-nearest neighbors issues

- Selecting K is always a problem
- What distance metric to use?
- What in case of hundreds of features?
  - Problemin analysing such a large number of parameters
  - Dimensionality curse
- It's local only

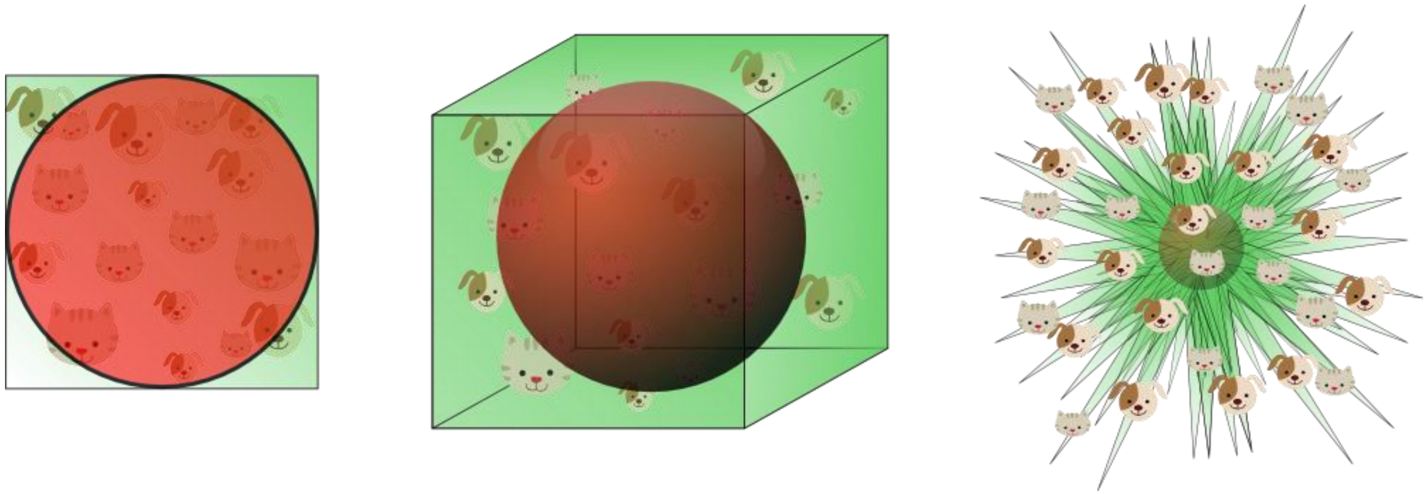


K-nearest neighbors issues

- Selecting K is always a problem
- What distance metric to use?
- What in case of hundreds of features?
  - Problemin analysing such a large number of parameters
  - Dimensionality curse
- It's local only

K-nearest neighbors issues

- Selecting K is always a problem
- What distance metric to use?
- What in case of hundreds of features?
  - Problemin analysing such a large number of parameters
  - Dimensionality curse
- It's local only



$$\frac{V_{\text{hypersphere}}}{V_{\text{hypercube}}} = \frac{\pi^{d/2}}{d2^{d-1}\Gamma(d/2)} \to 0, \text{as } d \to \infty$$

Logistic regression

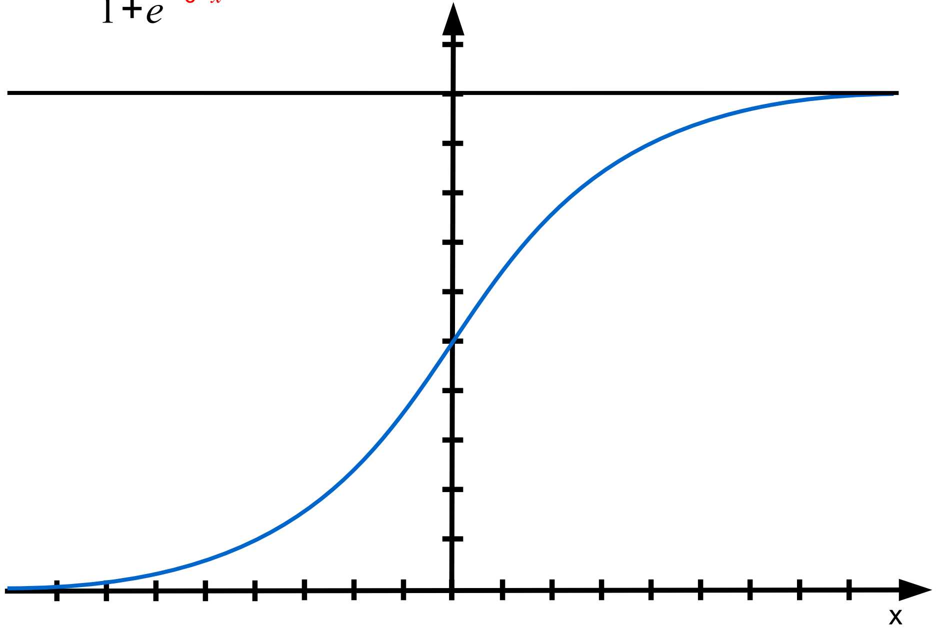
Logistic regression

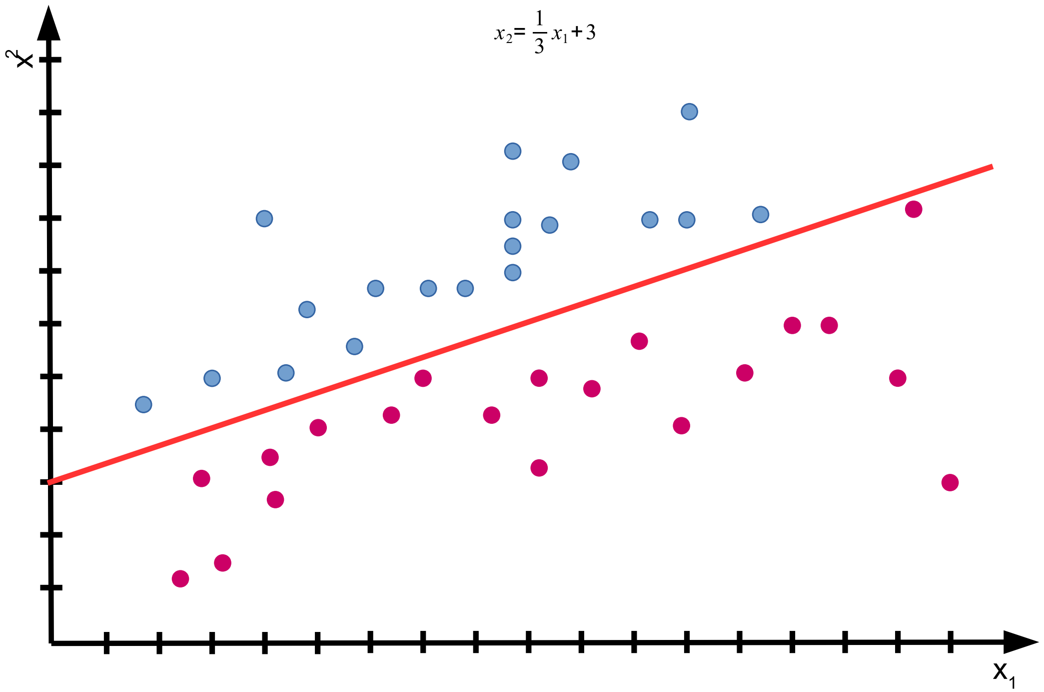
$$P(y^{(i)}=1)=\frac{1}{1+exp(-(\beta_0+\beta_1 x_1^{(i)}+\ldots+\beta_p x_p^{(i)}))}$$

$$\max_{\beta} ln \prod_{i=1}^{N} P(y^{(i)}|x^{(i)},\beta) = \underbrace{\max_{\beta} \sum_{i=1}^{N} lnP(y^{(i)}|x^{(i)},\beta)}_{\ell\ell(\beta)}$$

$$\max_{\beta} \ell\ell(\beta) = \max_{\beta} \sum_{i=1}^{N} \left[\mathbb{1}[y=+1]lnP(y^{(i)}=+1|x^{(i)},\beta)+\mathbb{1}[y=-1]lnP(y^{(i)}=-1|x^{(i)},\beta)\right]$$

$$f(x)=\frac{1}{1+e^{-\theta^T x}}=P(y=1|\theta,x)$$

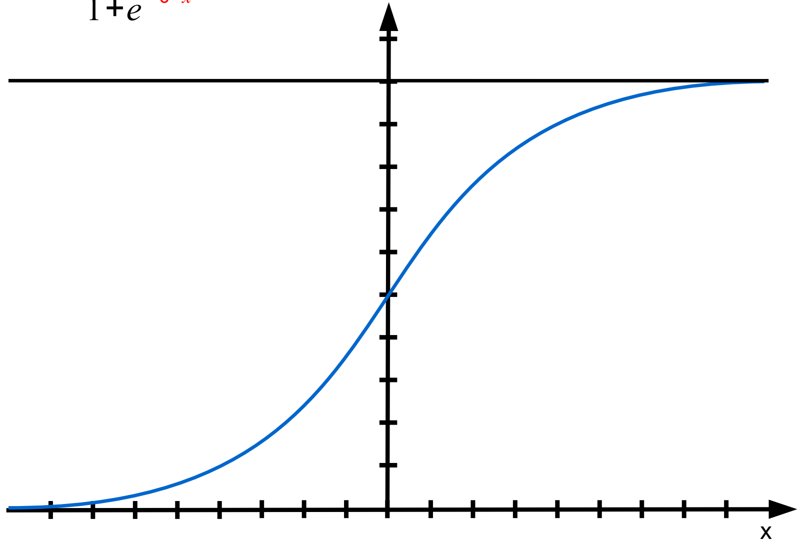




Logistic regression

- Interpreting numerical featrures
- Interpreting categorical features
- Normalization issue
- Feature importance

$$f(x) = \frac{1}{1+e^{-\theta^T x}} = P(y=1|\theta, x)$$



$$P(y^{(i)}=1) = \frac{1}{1+exp(-(\beta_0+\beta_1 x_1^{(i)}+\ldots+\beta_p x_p^{(i)}))}$$

$$ln\left(\frac{P(y=1)}{1-P(y=1)}\right) = log\left(\frac{P(y=1)}{P(y=0)}\right) = \beta_0+\beta_1 x_1+\ldots+\beta_p x_p$$

$$\frac{P(y=1)}{1-P(y=1)} = odds = exp\left(\beta_0+\beta_1 x_1+\ldots+\beta_p x_p\right)$$

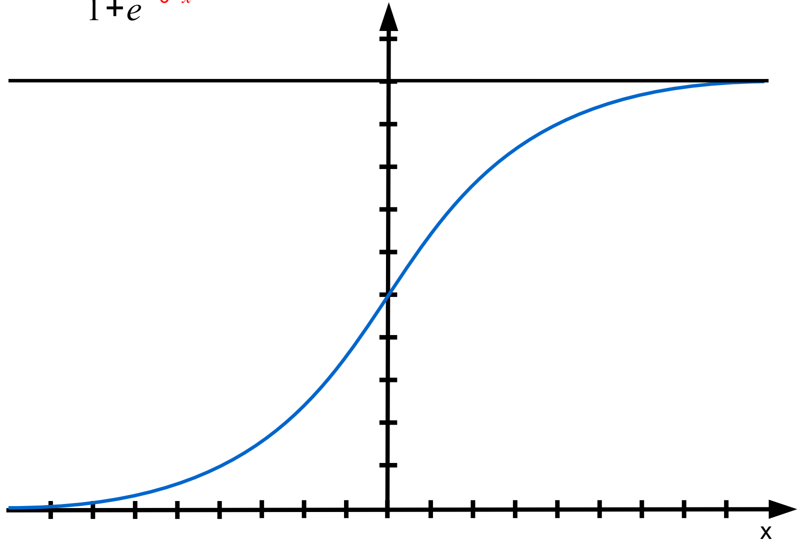
$$\frac{odds_{x_j+1}}{odds x_j} = \frac{exp\left(\beta_0+\beta_1 x_1+\ldots+\beta_j (x_j+1)+\ldots+\beta_p x_p\right)}{exp\left(\beta_0+\beta_1 x_1+\ldots+\beta_j x_j+\ldots+\beta_p x_p\right)}$$

$$\frac{odds_{x_j+1}}{odds x_j} = exp\left(\beta_j (x_j+1)-\beta_j x_j\right) = exp\left(\beta_j\right)$$

Logistic regression

- Interpreting numerical featrures
- Interpreting categorical features
- Normalization issue
- Feature importance

$$f(x) = \frac{1}{1+e^{-\theta^T x}} = P(y=1|\theta, x)$$



$$\pi_i = P(y_i = 1|x_i)$$

$$\mathbf{X} = \begin{bmatrix} 1 & x_{1,1} & \dots & x_{1,p} \\ 1 & x_{2,1} & \dots & x_{2,p} \\ \vdots & \vdots & \ddots & \vdots \\ 1 & x_{n,1} & \dots & x_{n,p} \end{bmatrix} \qquad \mathbf{V} = \begin{bmatrix} \hat{\pi}_1(1-\hat{\pi}_1) & 0 & \dots & 0 \\ 0 & \hat{\pi}_2(1-\hat{\pi}_2) & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & \hat{\pi}_n(1-\hat{\pi}_n) \end{bmatrix}$$

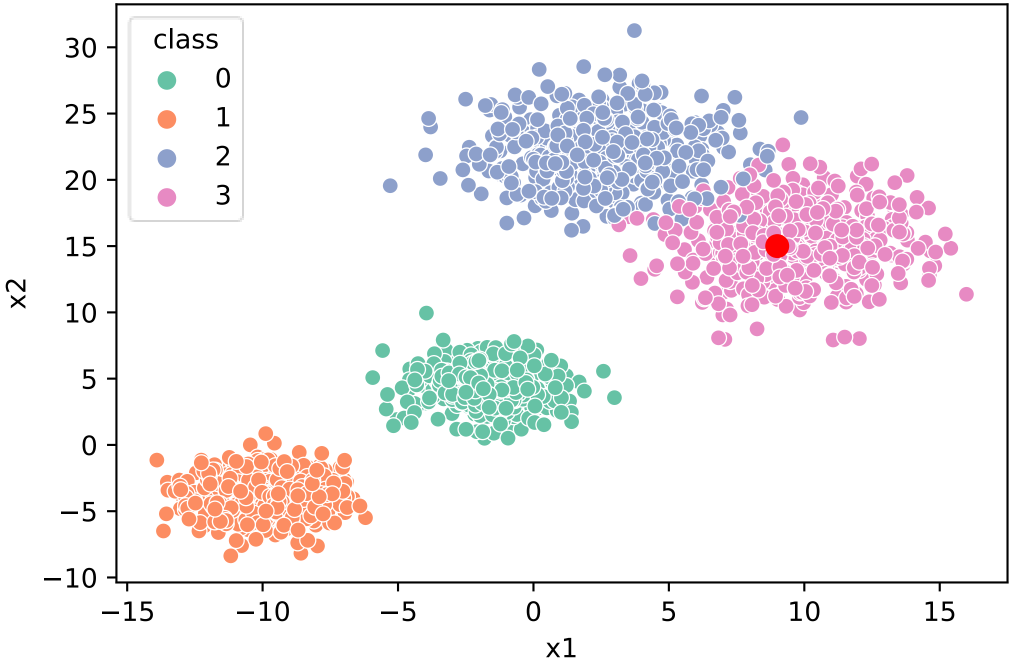
$$SE(\hat{\beta}) = (X^T V X)^{-1}$$

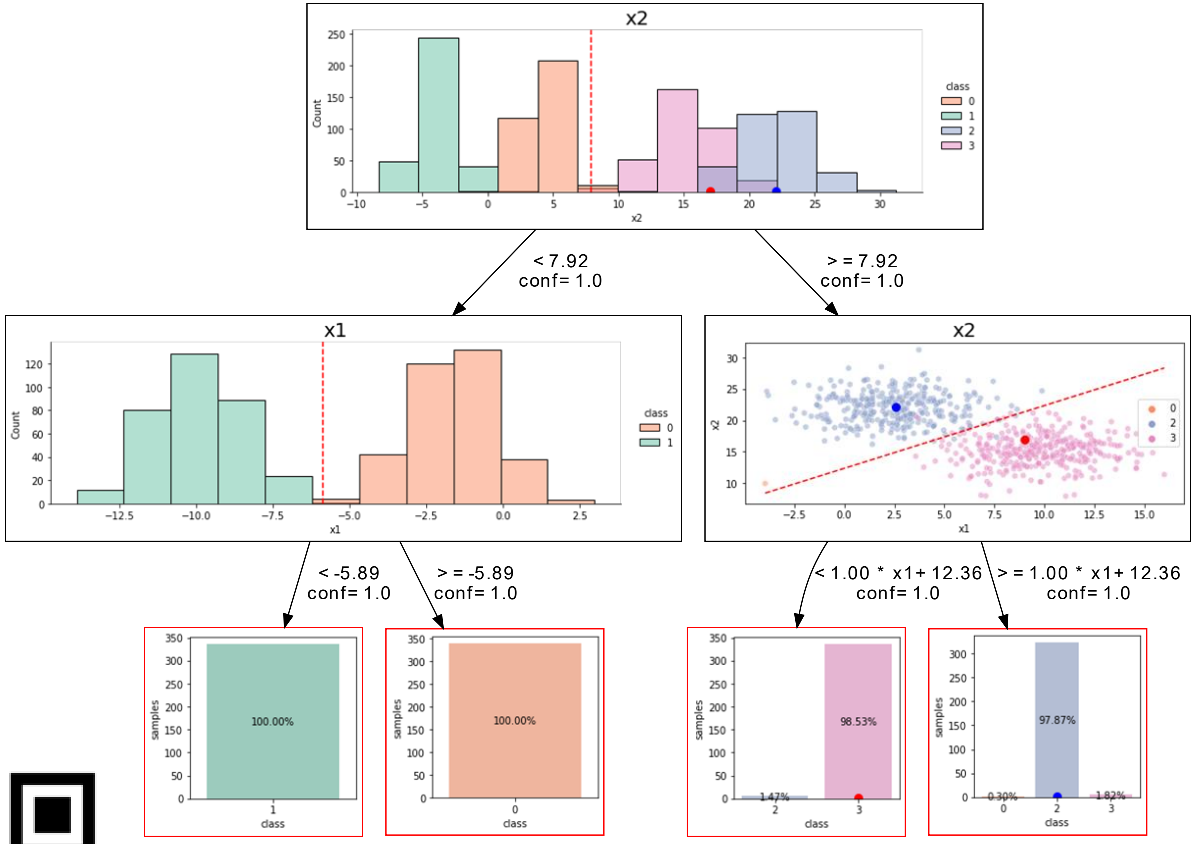
$$\frac{odds_{x_j+1}}{odds x_j} = exp\left(\beta_j(x_j+1) - \beta_j x_j\right) = exp\left(\beta_j\right)$$

$$t_{\hat{\beta}_j} = \frac{exp(\hat{\beta}_j)}{SE(\hat{\beta}_j))}$$

Decision trees

Decision trees





$$H(D) = -\sum_{c \in C} p(c) \log_2 p(c)$$

$$Gain(D) = H(D) - \sum_{v \in Values(F)} \frac{|D_v|}{|D|} H(D_v)$$

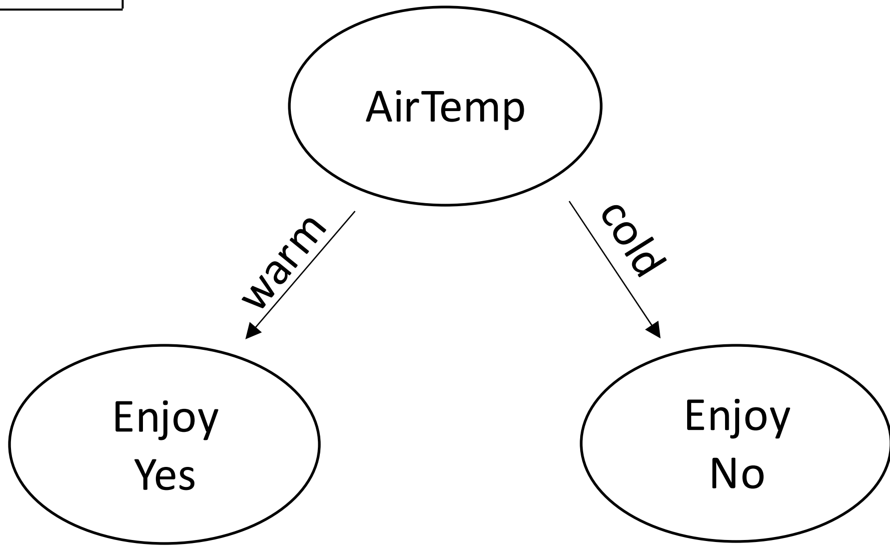
https://github.com/sbobek/lux

Decision trees

Outlook	AirTemp	Humidity	Windy	Water	Forecast	Enjoy
sunny	warm	normal	TRUE	warm	same	yes
sunny	warm	high	TRUE	warm	same	yes
rainy	cold	high	TRUE	warm	change	no
sunny	warm	high	TRUE	cool	change	yes
overcast	warm	normal	FALSE	warm	same	yes
overcast	cold	high	FALSE	cool	same	no

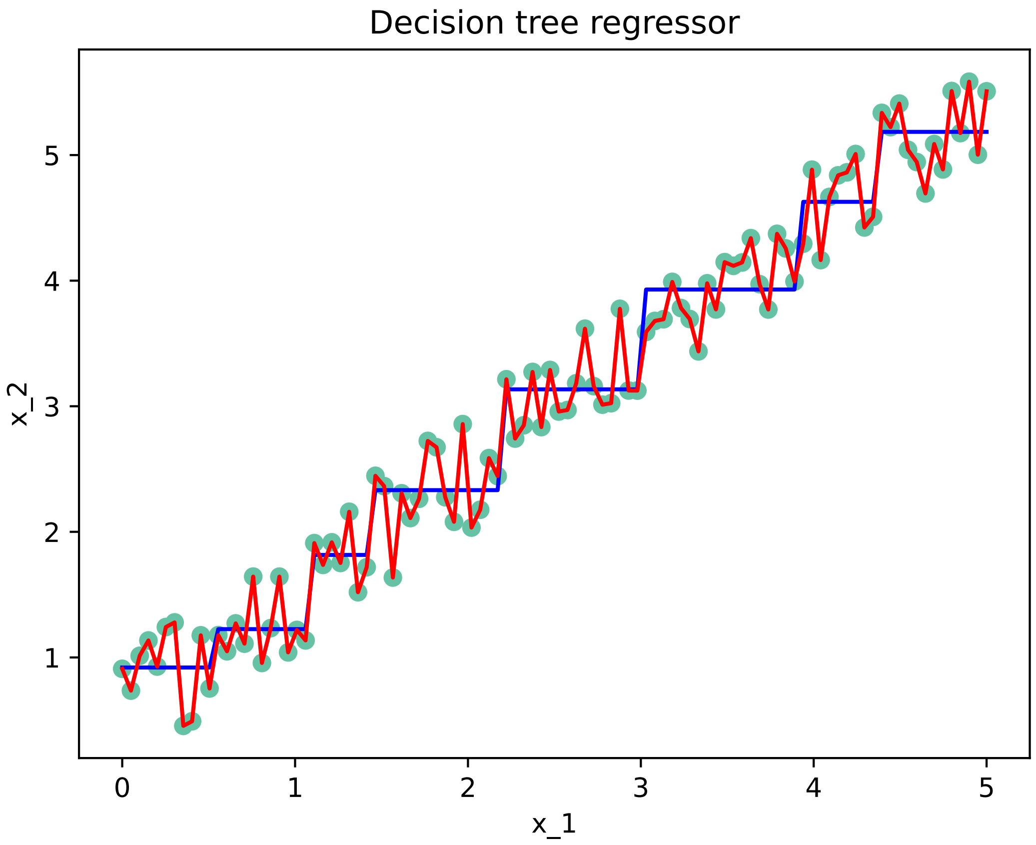
$$H(D) = -\sum_{c \in C} p(c) \log_2 p(c)$$

$$Gain(D) = H(D) - \sum_{v \in Values(F)} \frac{|D_v|}{|D|} H(D_v)$$



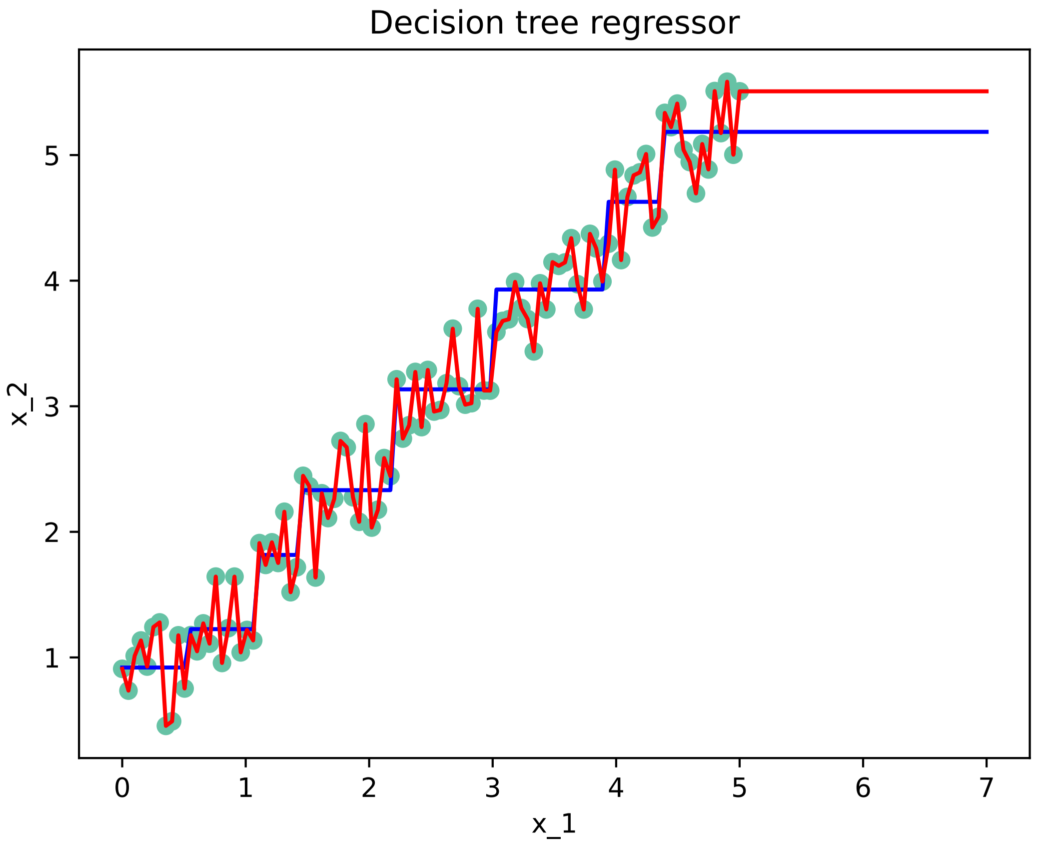
Pros and cons

- Nonparametric models – they are not that perfect for forecasting
- Can overfit without proper regularization
- No need to normalize/standarize/scale
- No need to One-hot-encode
- Feature importancecan be obtained immediatelly



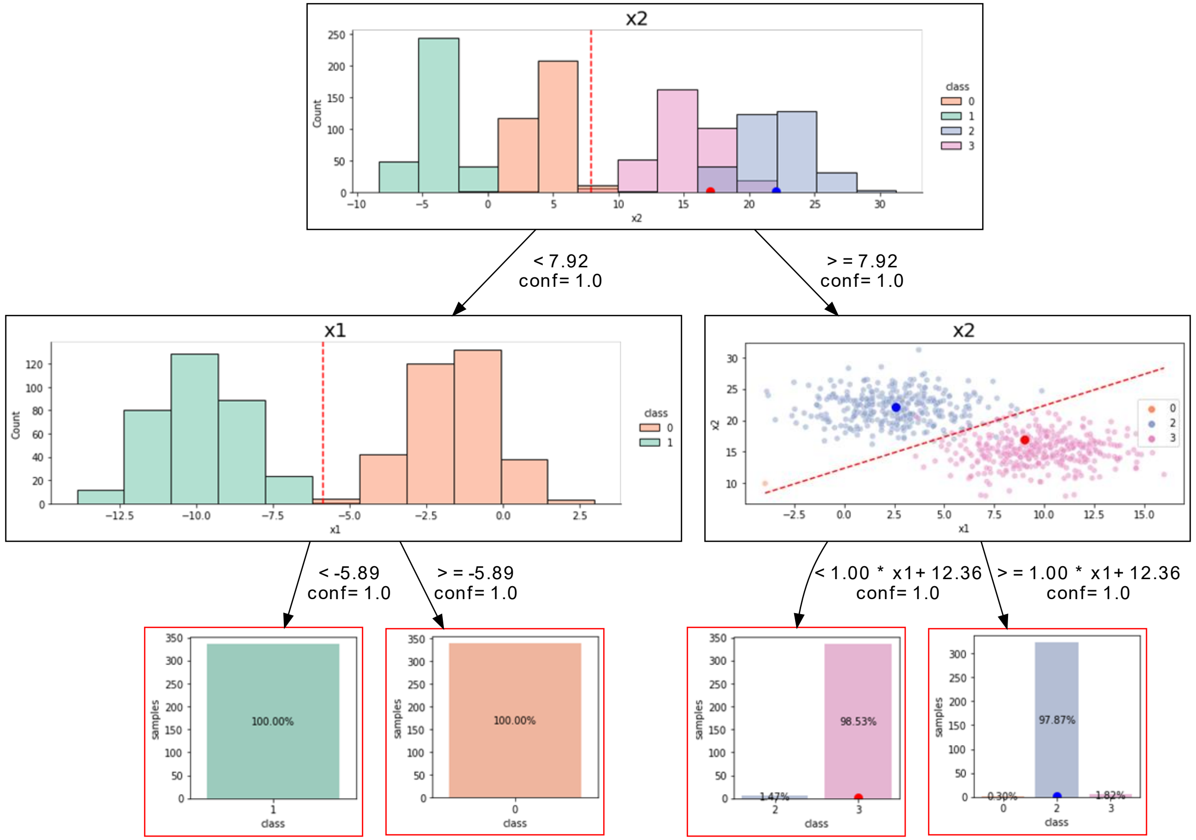
Pros and Cons

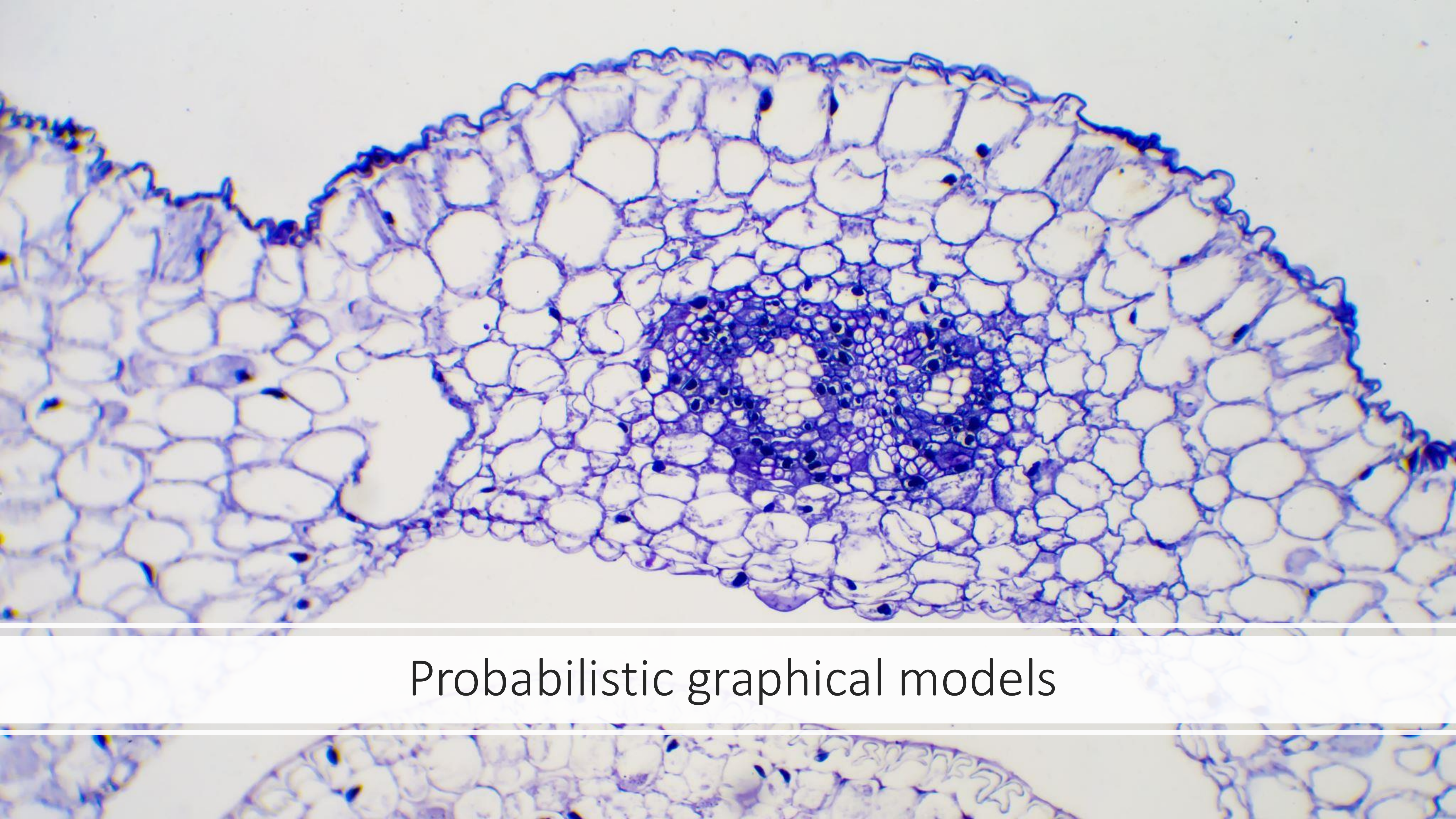
- Nonparametric models – they are not that perfect for forecasting
- Can overfit without proper regularization
- No need to normalize/standarize/scale
- No need to One-hot-encode
- Feature importancecan be obtained immediatelly



Pros and Cons

- Nonparametric models they are not that perfect for forefcasting
- Can overfit without proper regularization
- No need to normalize/standardize/scale
- No need to One-hot-encode
- Feature importanca can be obtained immediatelly

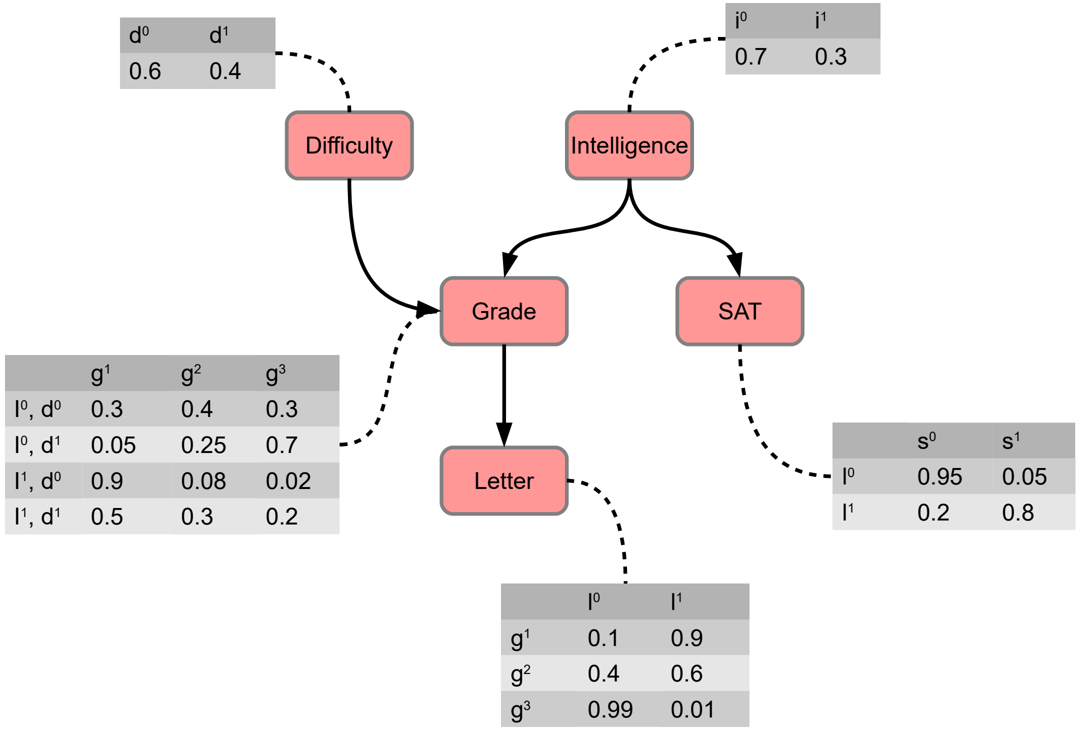




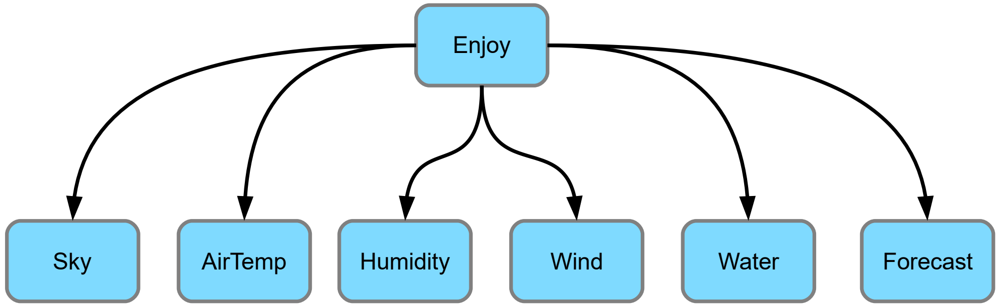
Probabilistic graphical models

Probabilistic graphical models (PGM)

- Nodes represent variables
- Edges represent direct probabilistic interactions
- Different types of PGM
  - o Bayesian entworks – acyclic, directed graphs
  - o Markov models – undirected graphs
- Easy incorporate domain knowledge
- Popular in causality modelling



Naive Bayes



Sky	AirTemp	Humidity	Wind	Water	Forecast	Enjoy
sunny	warm	normal	strong	warm	same	yes
sunny	warm	high	strong	warm	same	yes
rainy	cold	high	strong	warm	change	no
sunny	warm	high	strong	cool	change	no
cloudy	warm	normal	weak	warm	same	yes
cloudy	cold	high	weak	cool	same	no

- Conditional independence

$$P(Effect_1, \ldots, Effect_2|Cause) = P(Effect_1|Cause) \ldots P(Effect_n|Caues)$$

- Bayes rule

$$P(Cause|Effect_1, \ldots, Effect_n) = \frac{P(Cause)P(Effect_1, \ldots, Effect_n|Cause)}{P(Effect_1, \ldots, Effect_n)}$$

- Naive Bayes

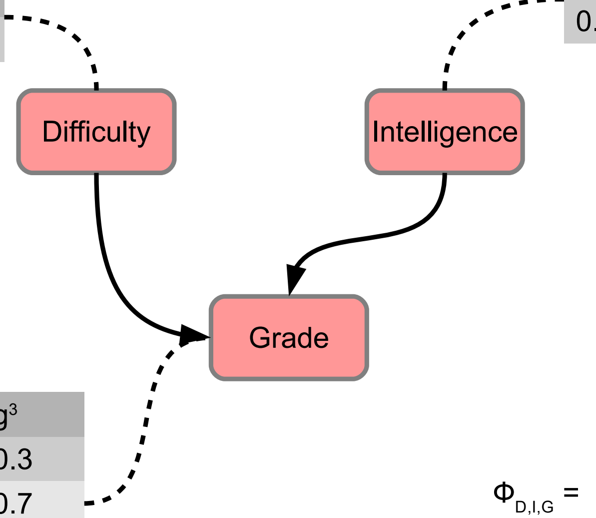
$$P(Cause|Effect_1, \ldots, Effect_n) = \alpha P(Cause) \prod_i P(Effect_i|Cause)$$

Inference in Bayesian Networks

- **Joint probability**
- Reduction
- Marginals
- MAP
- Tools for that

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

$\Phi_{D,I,G} =$

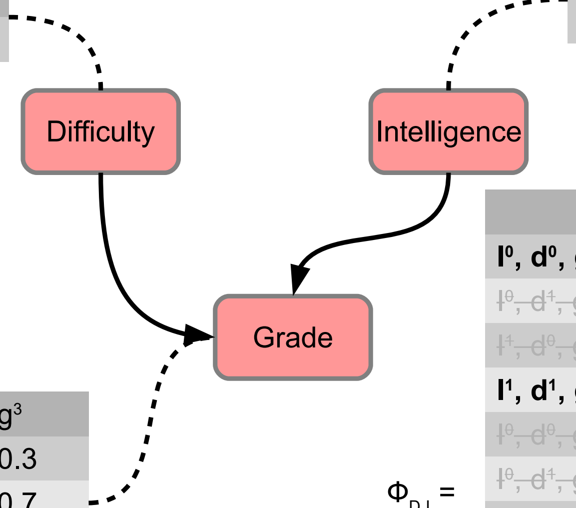
	P(D,I,G)
I^0, d^0, g^1	0.126
I^0, d^1, g^2	0.168
I^1, d^0, g^3	0.126
I^1, d^1, g^1	0.009
I^0, d^0, g^2	0.045
I^0, d^1, g^3	0.126
I^1, d^0, g^1	0.252
I^1, d^1, g^2	0.0224
I^0, d^0, g^3	0.0056
I^0, d^1, g^1	0.06
I^1, d^0, g^2	0.036
I^1, d^1, g^3	0.024
Sum	1.0

Inference in Bayesian Networks

- Joint probability
- **Reduction**
- Marginals
- MAP
- Tools for that

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

$\Phi_{D,I}$ =

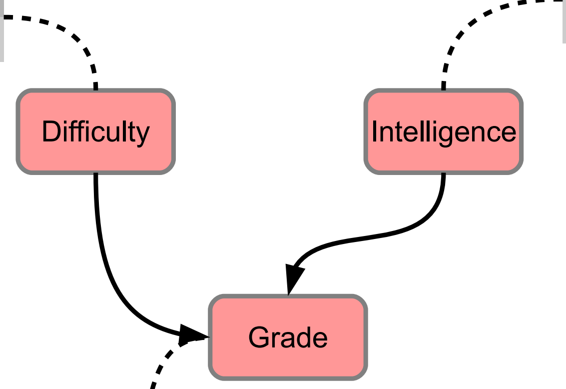
	$P(D,I \mid G=g^1)$
I^0, d^0, g^1	**0.126 / 0.447 = 0.282**
~~I^0, d^1, g^2~~	~~0.168~~
~~I^1, d^0, g^3~~	~~0.126~~
I^1, d^1, g^1	**0.02**
~~I^0, d^0, g^2~~	~~0.045~~
~~I^0, d^1, g^3~~	~~0.126~~
I^1, d^0, g^1	**0.564**
~~I^1, d^1, g^2~~	~~0.0224~~
~~I^0, d^0, g^3~~	~~0.0056~~
I^0, d^1, g^1	**0.134**
~~I^1, d^0, g^2~~	~~0.036~~
~~I^1, d^1, g^3~~	~~0.024~~
Sum	1.0

Inference in Bayesian Networks

- Joint probability
- Reduction
- **Marginals**
- MAP
- Tools for that

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

$\Phi_{D,G} = \Sigma$

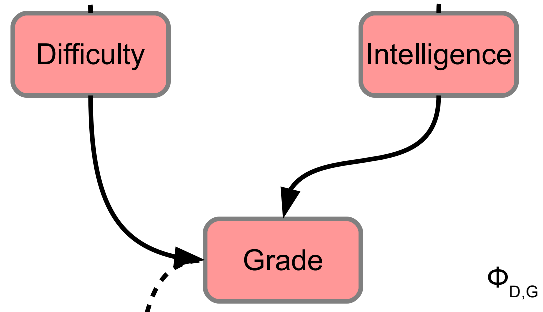
	P(D,G)
I^0, ~~d^0~~, g^1	**0.126**
I^0, ~~d^1~~, g^2	0.168
I^1, ~~d^0~~, g^3	0.126
I^1, ~~d^1~~, g^1	0.009
I^0, ~~d^0~~, g^2	0.045
I^0, ~~d^1~~, g^3	0.126
I^1, ~~d^0~~, g^1	0.252
I^1, ~~d^1~~, g^2	0.0224
I^0, ~~d^1~~, g^3	0.0056
I^0, ~~d^1~~, g^1	**0.06**
I^1, ~~d^0~~, g^2	0.036
I^1, ~~d^1~~, g^3	0.024
Sum	1.0

Inference in Bayesian Networks

- Joint probability
- Reduction
- **Marginals**
- MAP
- Tools for that

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

$\Phi_{D,G} =$

	P(D,G)
I^0, g^1	0.186
I^0, g^2	0.213
I^0, g^3	0.1316
I^1, g^1	0.261
I^1, g^2	0.0584
I^1, g^3	0.15
Sum	1.0

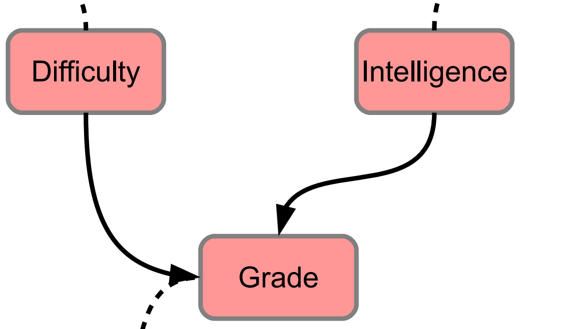
$$\sum_I P(I, D, G) = P(D, G)$$

Inference in Bayesian Networks

- Joint probability
- Reduction
- Marginals
- **MAP**
- Tools for that

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

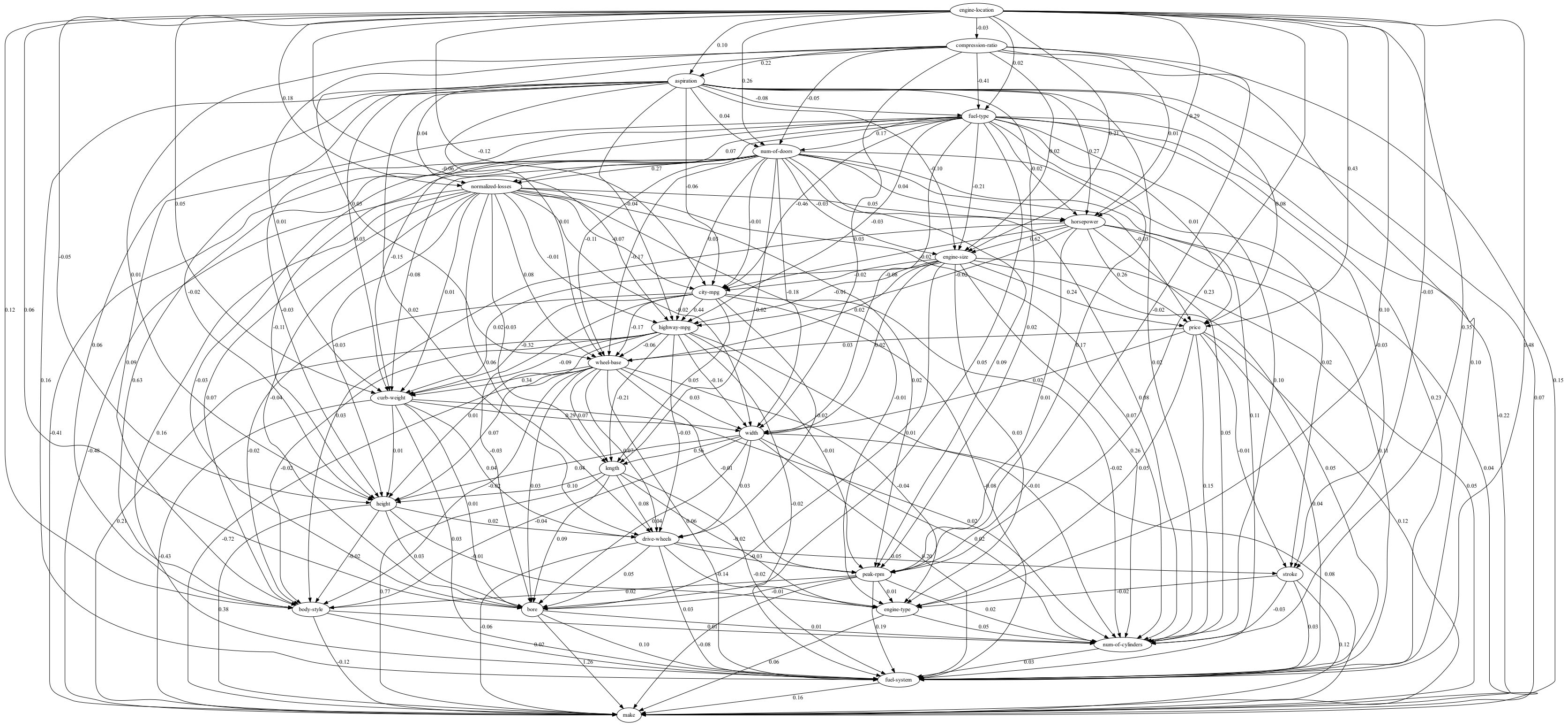
$\Phi_{D,I,G} =$

	P(D,I,G)
I^0, d^0, g^1	0.126
I^0, d^1, g^2	0.168
I^1, d^0, g^3	0.126
I^1, d^1, g^1	0.009
I^0, d^0, g^2	0.045
I^0, d^1, g^3	0.126
I^1, d^0, g^1	0.252
I^1, d^1, g^2	0.0224
I^0, d^0, g^3	0.0056
I^0, d^1, g^1	0.06
I^1, d^0, g^2	0.036
I^1, d^1, g^3	0.024
Sum	1.0

$$MAP_D = argmax_{i,d,g} P(I, D, G)$$

Inference in Bayesian Networks

- Joint probability
- Reduction
- Marginals
- MAP
- **Tools for that**



Note: In real-life examples exact inference is not an option (usually). Additionally, we need tools that will help us learn the structure, lenrn CPODs, manage large networks, etc.

Tools for BN Leanring and inference

- PGMPy
- **CausalNEX**
- DoWhy
- Pyro
- **ProbLog**
- …

```
evidence(contains_word(money), true).
evidence(contains_word(discount), false).
evidence(contains_word(winner), false).
evidence(from_unknown_sender, true).
evidence(contains_attachment, false).
query(spam).
% Response:
% spam:   0.216
% There's a 21.6% chance that emails with
% these features is a spam
```

```
0.2::spam.
0.4::contains_word(money) :- spam.
0.5::contains_word(discount) :- spam.
0.7::contains_word(winner) :- spam.
0.3::from_unknown_sender :- spam.
0.1::contains_attachment :- spam.

0.6::contains_word(money) :- not(spam).
0.5::contains_word(discount) :- not(spam).
0.3::contains_word(winner) :- not(spam).
0.7::from_unknown_sender :- not(spam).
0.9::contains_attachment :- not(spam).
```

```
from sklearn.model_selection import train_test_split
train test = train_test_split(discretised_data, train_size=0.9, test_size=0.1, random_state=7)
bn = bn.fit_node_states(discretised_data)
bn = bn.fit_cpds(train, method="BayesianEstimator", bayes_prior="K2")

from causalnex.inference import InferenceEngine
ie = InferenceEngine(bn)
marginals_short = ie.query({"studytime": "short-studytime"})
marginals_long = ie.query({"studytime": "long-studytime"})
print("Marginal G1 | Short Studytime", marginals_short["G1"])
print("Marginal G1 | Long Studytime", marginals_long["G1"])
---
Marginal G1 | Short Studytime {'Fail': 0.2776556433482524, 'Pass':
0.7223443566517477}
Marginal G1 | Long Studytime {'Fail': 0.15504850337837614, 'Pass':
```

Thank you for your attention!

JAGIELLONIAN UNIVERSITY
IN KRAKÓW

https://geist.re