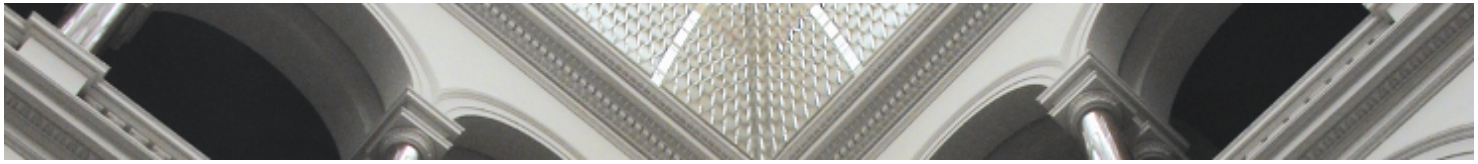
Technische Universität Berlin

Classical & Linear Methods #1: Regression & Regularization

Seul-Ki Yeom
Technische Universität Berlin - Machine Learning Group
Beginners Machine Learning Workshop 2019



Technische Universität Berlin

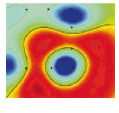
Agenda

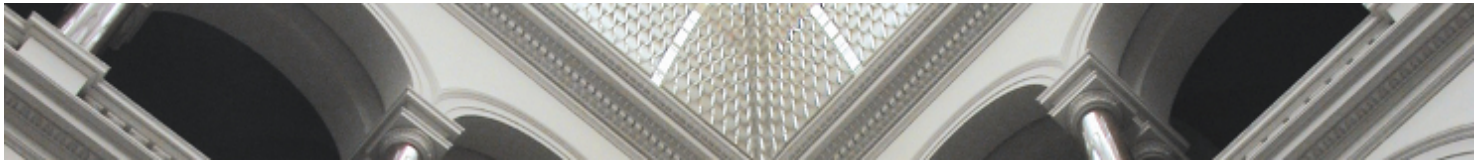
Linear Regression

Regularization & Ridge Regression

Logistic Regression

Example: Prothesis control





Technische Universität Berlin

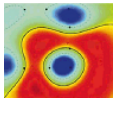
Agenda

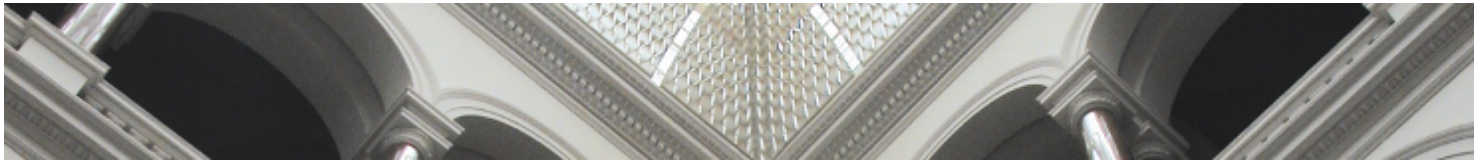
Linear Regression

Regularization & Ridge Regression

Logistic Regression

Example: Prothesis control





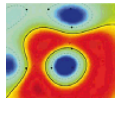
Covariance and Correlation

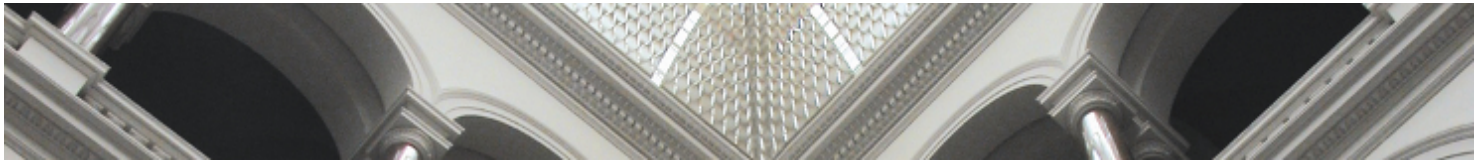
For two random variables X and Y, their **covariance** and **correlation** are defined as

$$\mathrm{Cov}(X, Y) := E[(X - E(X))(Y - E(Y))] = \frac{1}{N-1}\sum_{i=1}^{N}(X_i - \overline{X})(Y_i - \overline{Y})$$

$$\mathrm{Corr}(X, Y) := \frac{\mathrm{Cov}(X, Y)}{\sqrt{\mathrm{Var}(X)}\sqrt{\mathrm{Var}(Y)}}$$

- Both of these two determine the relationship and measures the dependency between two random variables.
- **Covariance** is a value for measuring of the variance between two variables.
- **Correlation** is the normalized covariance. **Correlation** is a value when the change in one variable may result in the change in the another variable.





Technische Universität Berlin

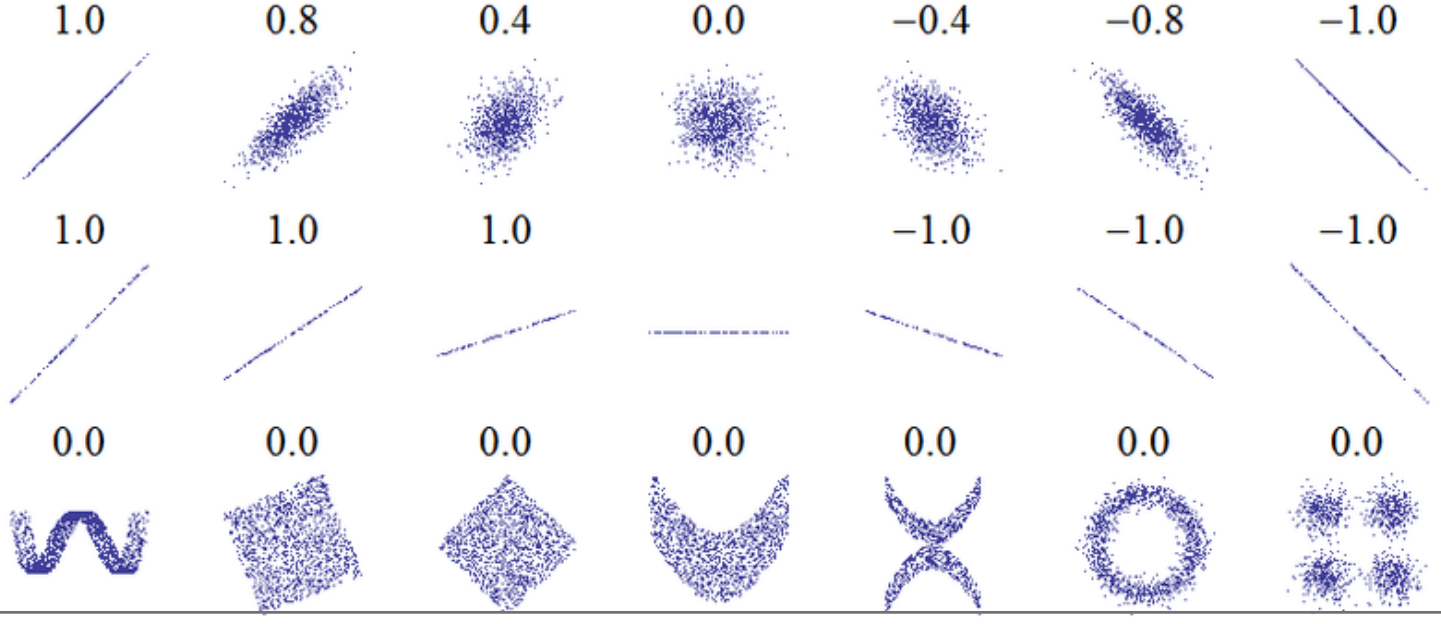
Covariance and Correlation

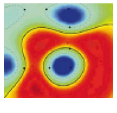
For two random variables X and Y, their **covariance** and **correlation** are defined as

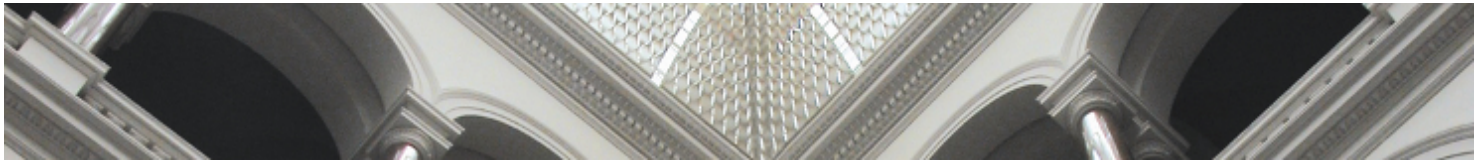
$$\mathrm{Cov}(X, Y) := E[(X - E(X))(Y - E(Y))] = \frac{1}{N-1}\sum_{i=1}^{N}(X_i - \overline{X})(Y_i - \overline{Y})$$

$$\mathrm{Corr}(X, Y) := \frac{\mathrm{Cov}(X, Y)}{\sqrt{\mathrm{Var}(X)}\sqrt{\mathrm{Var}(Y)}}$$

Correlation measures the linear relationship between X and Y:







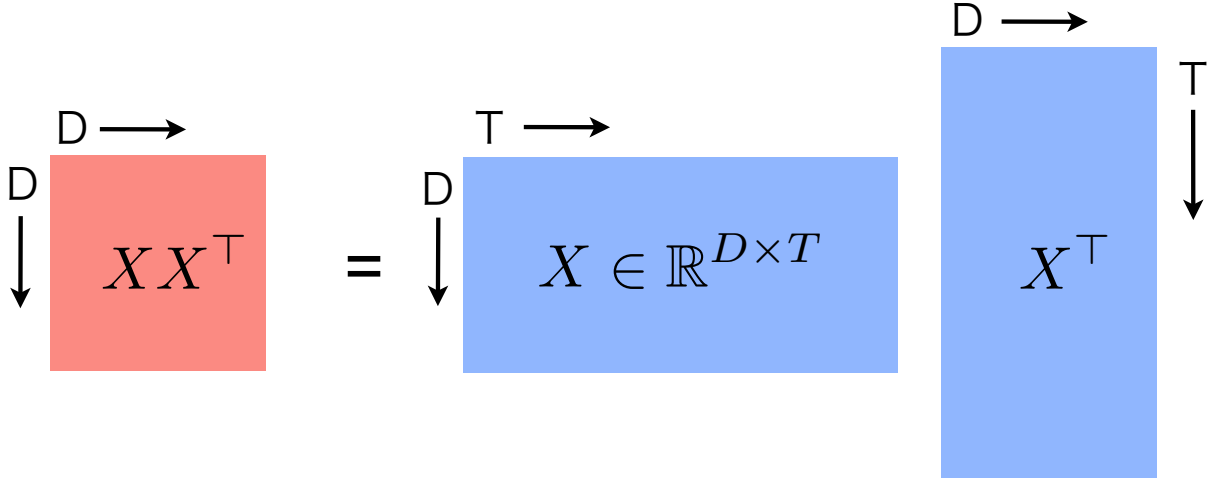
Technische Universität Berlin

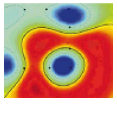
Covariance Matrices

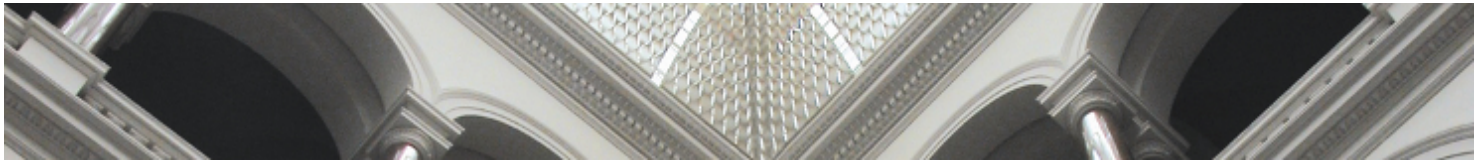
Given T data points $\mathbf{x}_t \in \mathbb{R}^D$ in a data matrix $X \in \mathbb{R}^{D \times T}$ the empirical estimate of the **covariance matrix** is defined as

$$S = \frac{1}{T} XX^\top \tag{1}$$

where we assume centered data, i.e. $\sum_{t=1}^{T} \mathbf{x}_t = \mathbf{0}$.



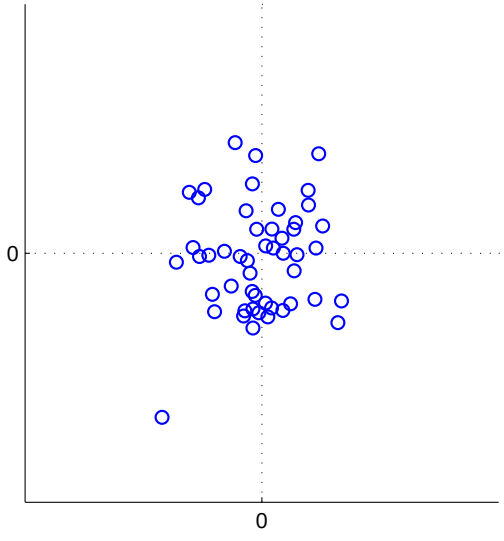




Correlated Data and Linear Mappings

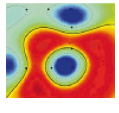
We can generate correlated data using a diagonal scaling matrix D and a rotation R

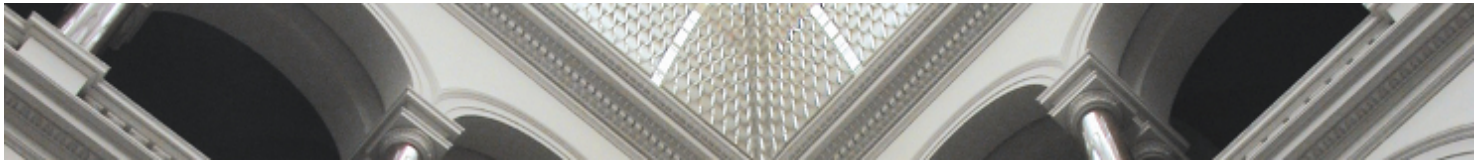
Uncorrelated



$$x \sim \mathcal{N}(0, 1)$$

$$XX^\top = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$$

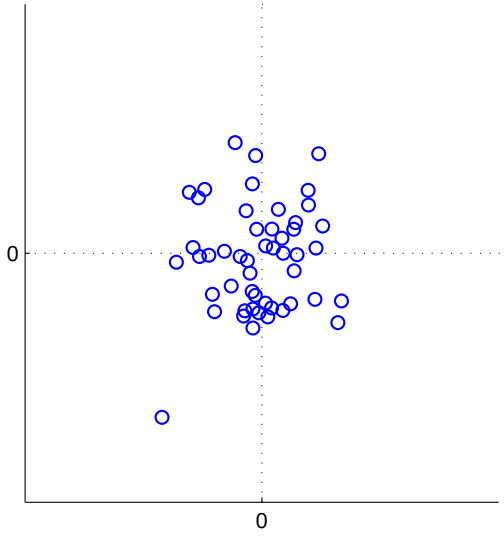




Correlated Data and Linear Mappings

We can generate correlated data using a diagonal scaling matrix D and a rotation R

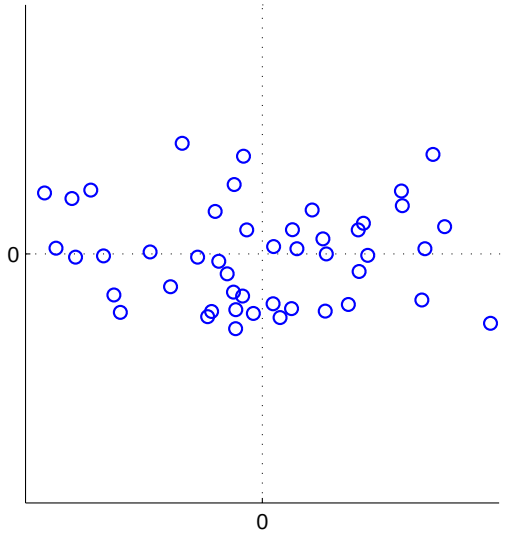
Uncorrelated



$$x \sim \mathcal{N}(0, 1)$$

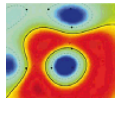
$$xx^\top = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$$

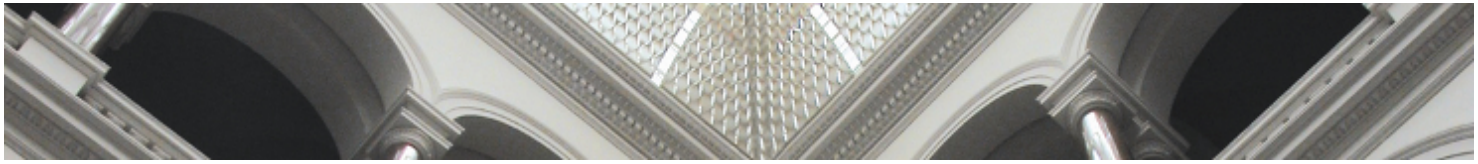
Uncorrelated, scaled



$$X = \begin{bmatrix} 3 & 0 \\ 0 & 1 \end{bmatrix}$$

$$xx^\top = \begin{bmatrix} 9 & 0 \\ 0 & 1 \end{bmatrix}$$

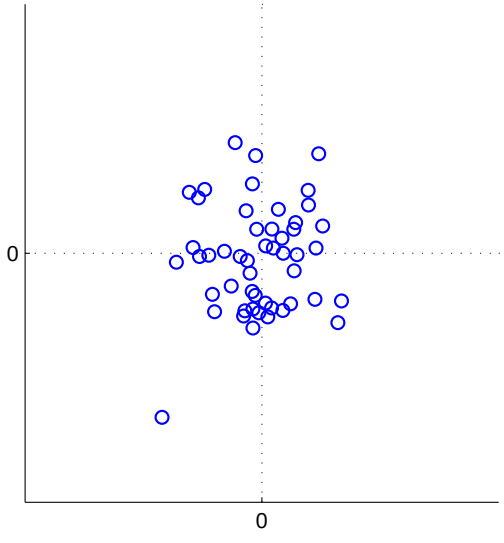




Correlated Data and Linear Mappings

We can generate correlated data using a diagonal scaling matrix D and a rotation R

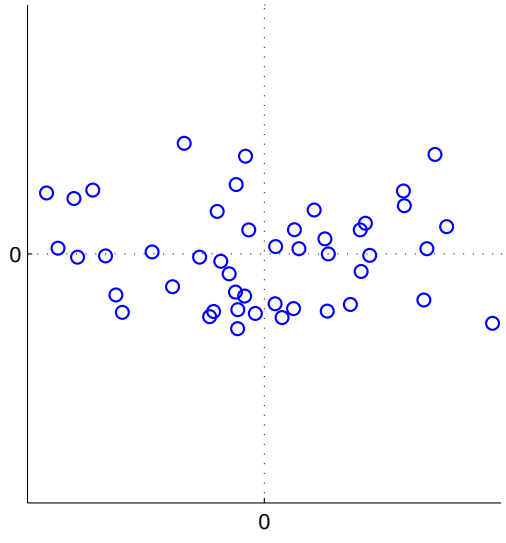
Uncorrelated



$$x \sim \mathcal{N}(0, 1)$$

$$XX^\top = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$$

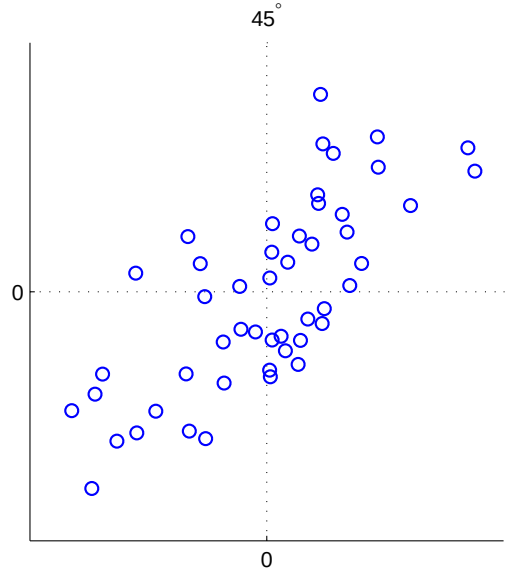
Uncorrelated, scaled



$$X = \begin{bmatrix} 3 & 0 \\ 0 & 1 \end{bmatrix}$$

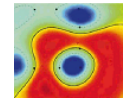
$$XX^\top = \begin{bmatrix} 9 & 0 \\ 0 & 1 \end{bmatrix}$$

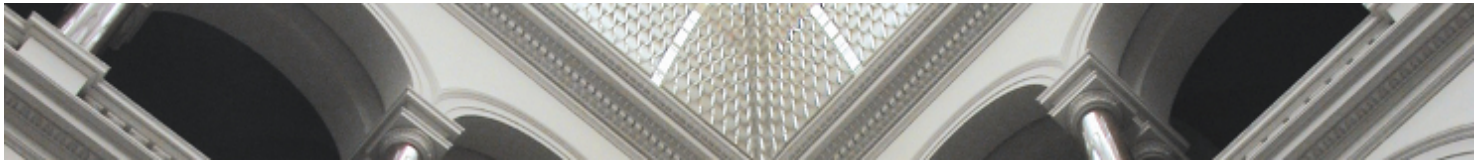
Scaled, rotated by 45°



$$\begin{bmatrix} cos(\phi) & -sin(\phi) \\ sin(\phi) & cos(\phi) \end{bmatrix} X$$

$$XX^\top = \begin{bmatrix} 5 & 4 \\ 4 & 5 \end{bmatrix}$$

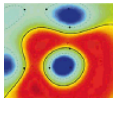


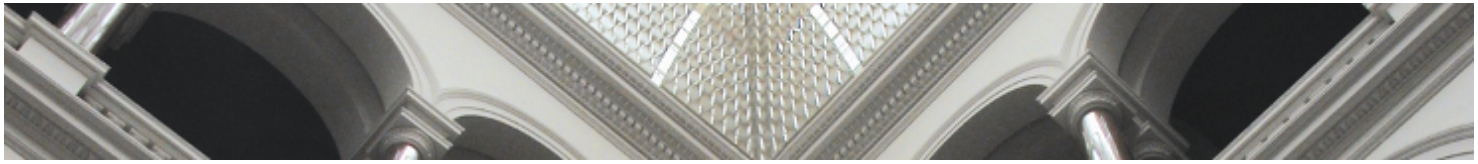


Technische Universität Berlin

Linear Regression - Application examples

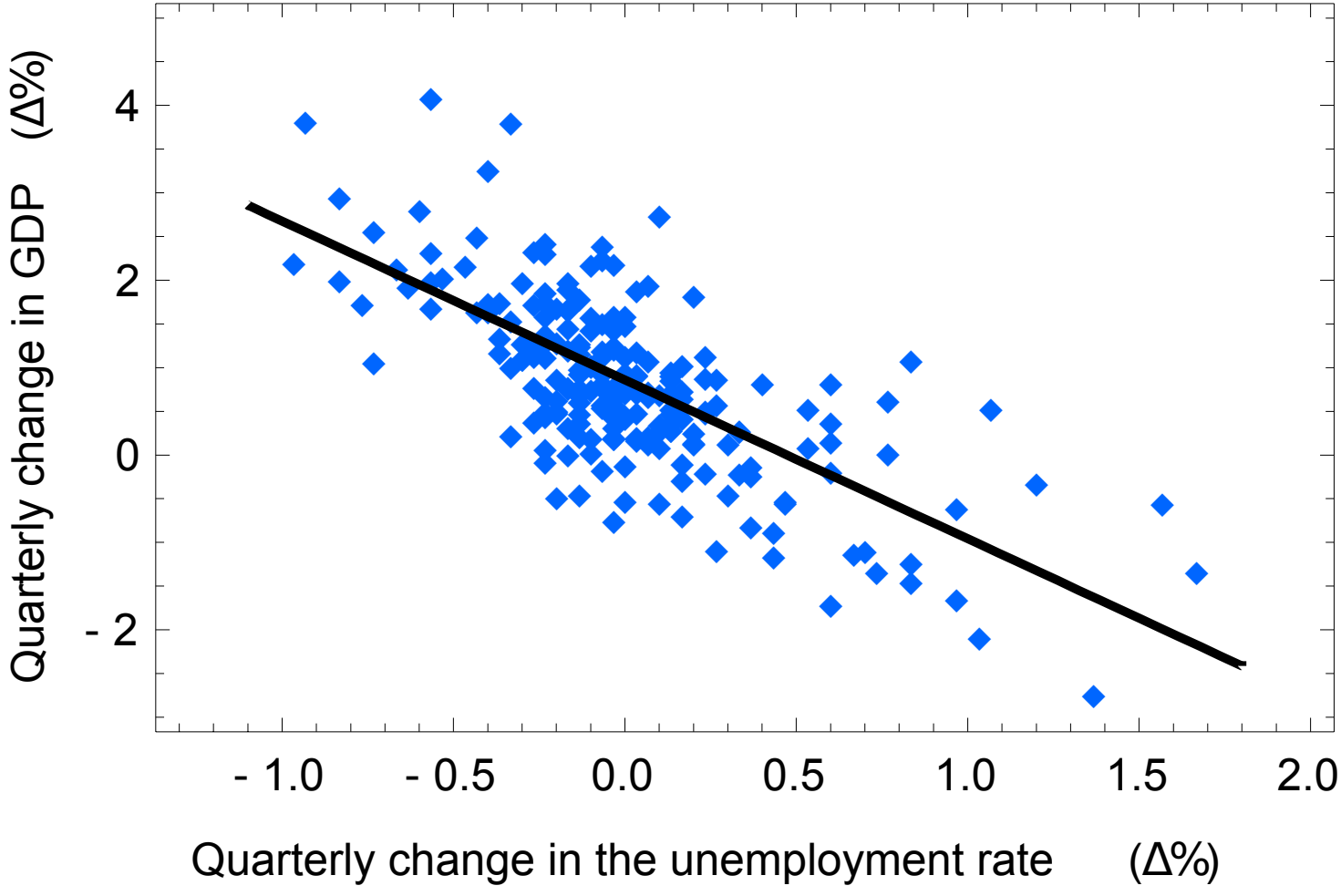
- Predict BMW stock price in 6 months based on Germany economic data and company performance measures
- Predict crop yield from weather variables
- Control a hand prothesis based on electric activity measured on the arm
- . . .

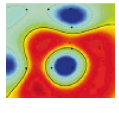


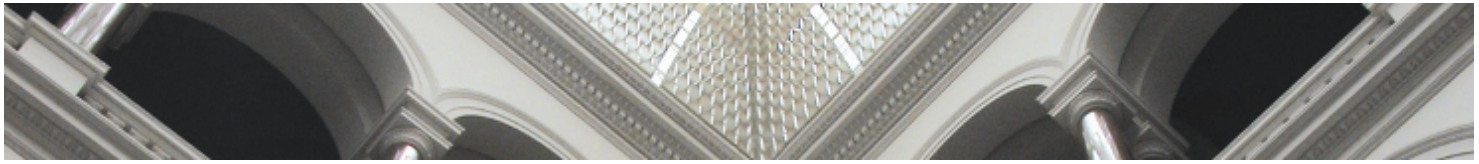


Technische Universität Berlin

Simple Linear Regression





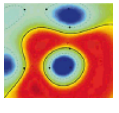


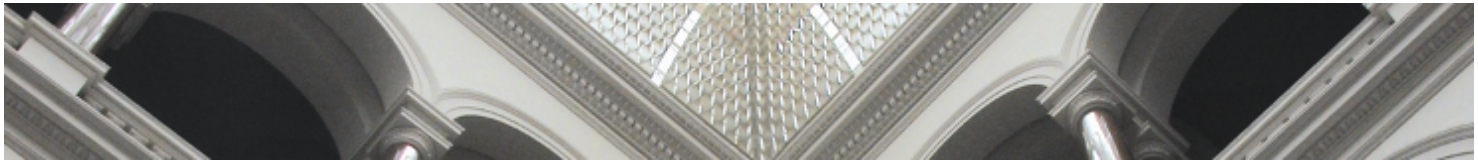
Simple Linear Regression

Given data $x_1, \ldots x_T \in \mathbb{R}$ and labels $y_1, \ldots, y_T \in \mathbb{R}$, the goal is to predict y by a linear function of x

$$y_t = \omega \cdot x_t$$

Q. How can we find the best linear function?





Technische Universität Berlin

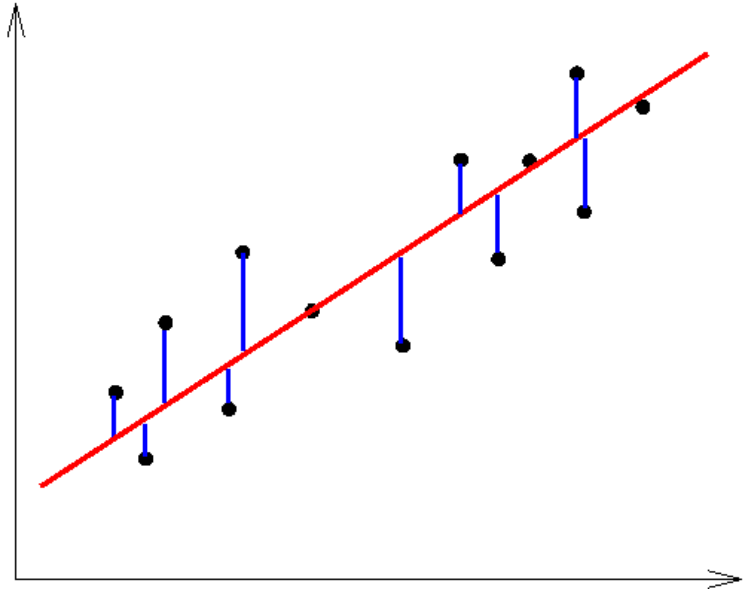
Simple Linear Regression

Given data $x_1, \ldots x_T \in \mathbb{R}$ and labels $y_1, \ldots, y_T \in \mathbb{R}$, the goal is to predict y by a linear function of x

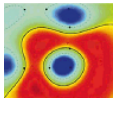
$$y_t = \omega \cdot x_t$$

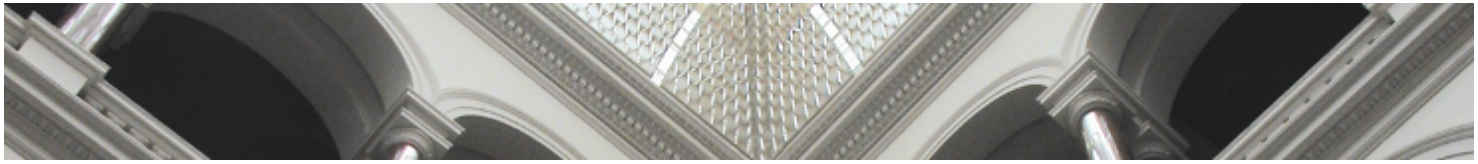
Q. How can we find the best linear function?

$\Rightarrow$ Minimize **least-square error** to find the ”best fit ” line to our (training) data



$$\mathcal{E}(\omega) = \sum_{t=1}^{T} (y_t - \omega \cdot x_t)^2$$





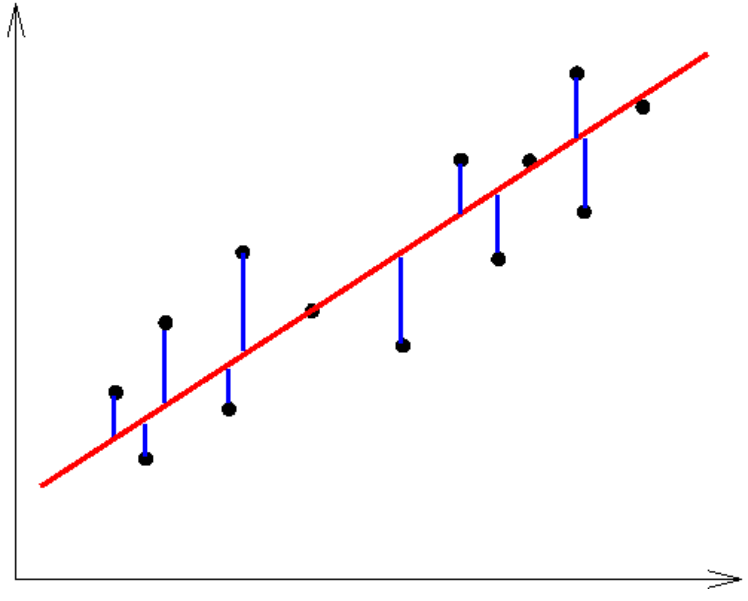
Simple Linear Regression

Given data $x_1, \ldots x_T \in \mathbb{R}$ and labels $y_1, \ldots, y_T \in \mathbb{R}$, the goal is to predict y by a linear function of x

$$y_t = \omega \cdot x_t$$

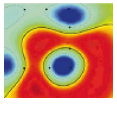
Q. How can we find the best linear function?

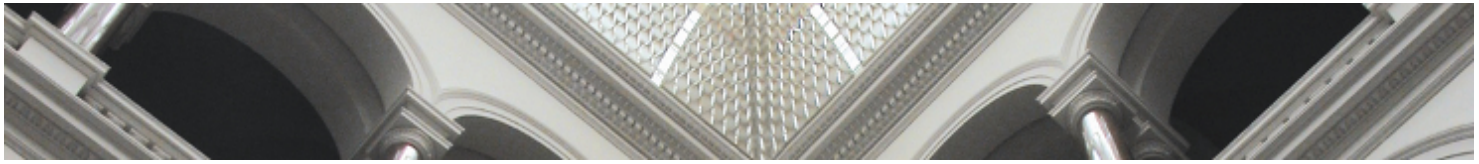
$\Rightarrow$ Minimize **least-square error** to find the ”best fit ” line to our (training) data



$$\mathcal{E}(\omega) = \sum_{t=1}^{T} (y_t - \omega \cdot x_t)^2$$

- optimal under Gaussian noise assumption
- differentiable
- leads to unique solution





Technische Universität Berlin

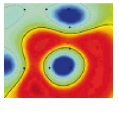
Simple Linear Regression

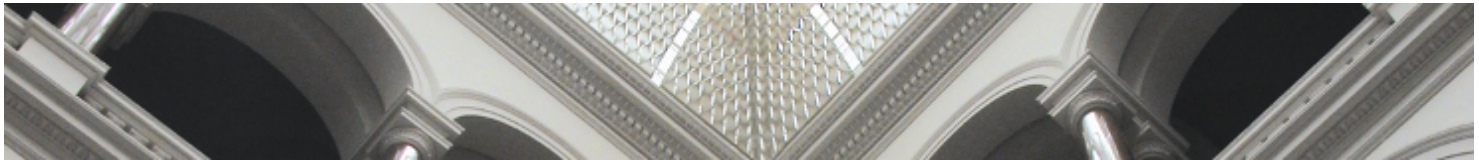
$$\mathcal{E}(\omega) = \sum_{t=1}^{T} (y_t - \omega \cdot x_t)^2$$

Q. What our function $\mathcal{E}(\omega)$ looks like?

x	y
1	1
2	2
3	3

- If $\omega = 1$, $\mathcal{E}(\omega)$=?





Technische Universität Berlin

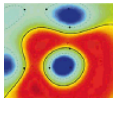
Simple Linear Regression

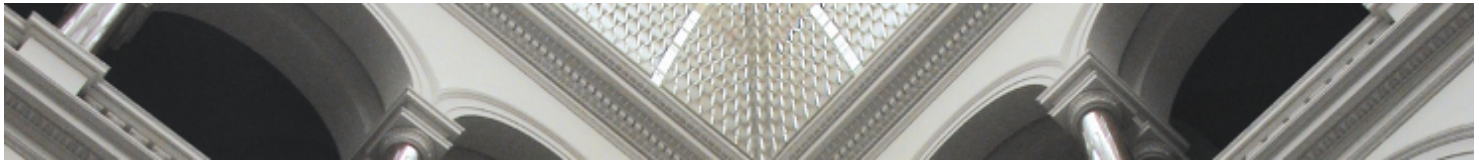
$$\mathcal{E}(\omega) = \sum_{t=1}^{T} (y_t - \omega \cdot x_t)^2$$

Q. What our function $\mathcal{E}(\omega)$ looks like?

x	y
1	1
2	2
3	3

- If $\omega = 1$, $\mathcal{E}(\omega)$=0
  $(1\text{-}1\times1)^2 + (2\text{-}1\times2)^2 + (3\text{-}1\times3)^2$
- If $\omega = 0$, $\mathcal{E}(\omega)$=14
  $(1\text{-}0\times1)^2 + (2\text{-}0\times2)^2 + (3\text{-}0\times3)^2$
- If $\omega = 2$, $\mathcal{E}(\omega)$=?



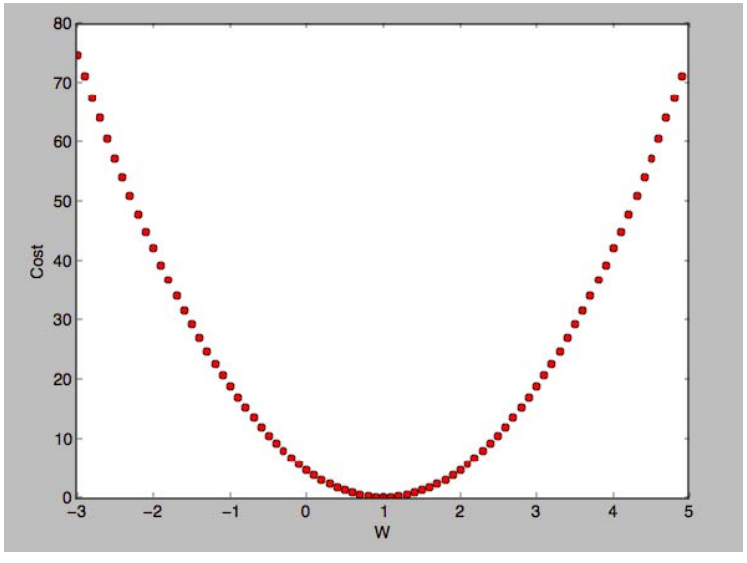


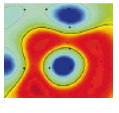
Technische Universität Berlin

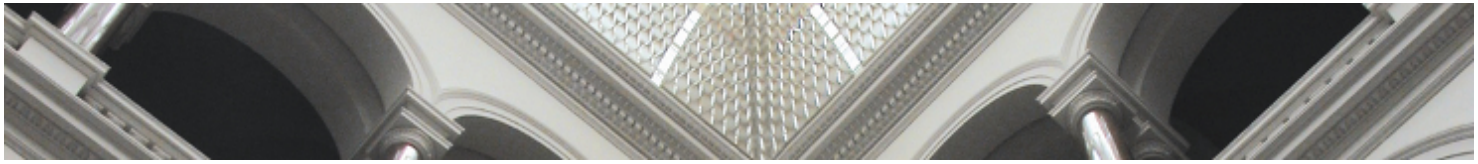
Simple Linear Regression

$$\mathcal{E}(\omega) = \sum_{t=1}^{T} (y_t - \omega \cdot x_t)^2$$

Q. What our function $\mathcal{E}(\omega)$ looks like?







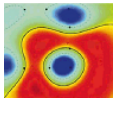
Technische Universität Berlin

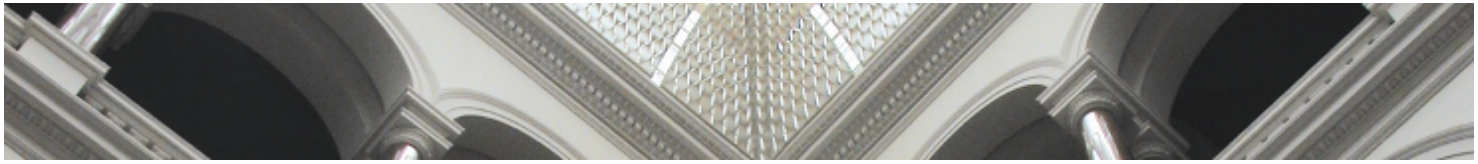
Simple Linear Regression

$$\mathcal{E}(\omega) = \sum_{t=1}^{T} (y_t - \omega \cdot x_t)^2$$

Compute the derivative w.r.t. ω

$$\frac{\partial \mathcal{E}(\omega)}{\partial \omega} = \sum_{t=1}^{T} 2(y_t - \omega \cdot x_t) \cdot (-x_t)$$





Technische Universität Berlin

Simple Linear Regression

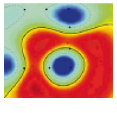
$$\mathcal{E}(\omega) = \sum_{t=1}^{T} (y_t - \omega \cdot x_t)^2$$

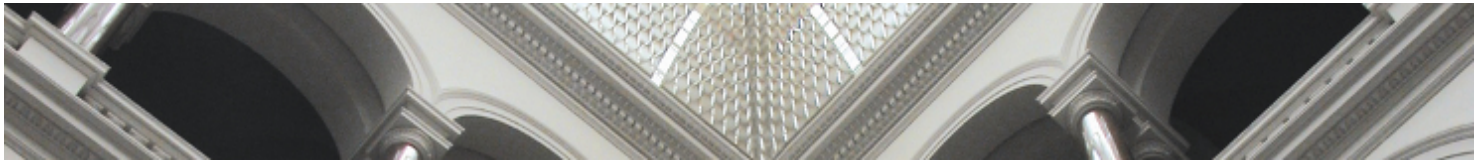
Compute the derivative w.r.t. ω

$$\frac{\partial \mathcal{E}(\omega)}{\partial \omega} = \sum_{t=1}^{T} 2(y_t - \omega \cdot x_t) \cdot (-x_t)$$

sets it to zero and solves for ω:

$$\sum_{t=1}^{T} 2(y_t - \omega \cdot x_t) \cdot (-x_t) = 0 \quad \Longrightarrow \quad \sum_{t=1}^{T} y_t x_t - \omega \sum_{t=1}^{T} x_t^2 = 0$$





Technische Universität Berlin

Simple Linear Regression

$$\mathcal{E}(\omega) = \sum_{t=1}^{T} (y_t - \omega \cdot x_t)^2$$

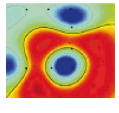
Compute the derivative w.r.t. ω

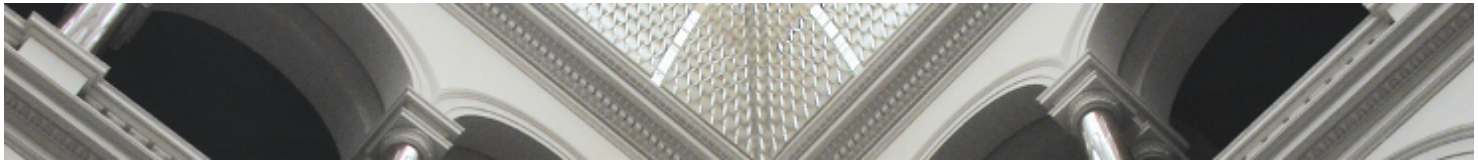
$$\frac{\partial \mathcal{E}(\omega)}{\partial \omega} = \sum_{t=1}^{T} 2(y_t - \omega \cdot x_t) \cdot (-x_t)$$

sets it to zero and solves for ω:

$$\sum_{t=1}^{T} 2(y_t - \omega \cdot x_t) \cdot (-x_t) = 0 \quad \Longrightarrow \quad \sum_{t=1}^{T} y_t x_t - \omega \sum_{t=1}^{T} x_t^2 = 0$$

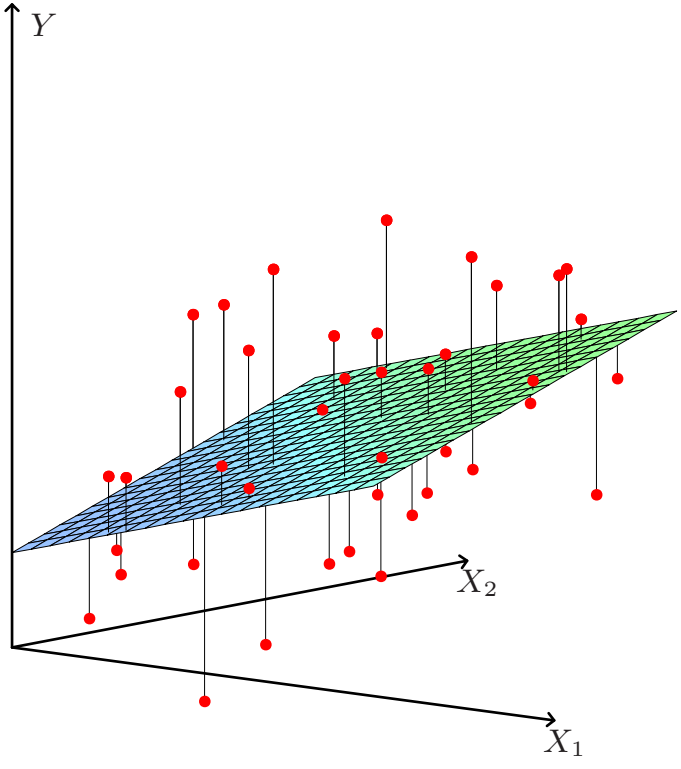
$$\Longrightarrow \quad \omega = \frac{\sum_{t=1}^{T} x_t y_t}{\sum_{t=1}^{T} x_t x_t}$$



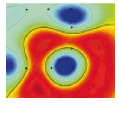


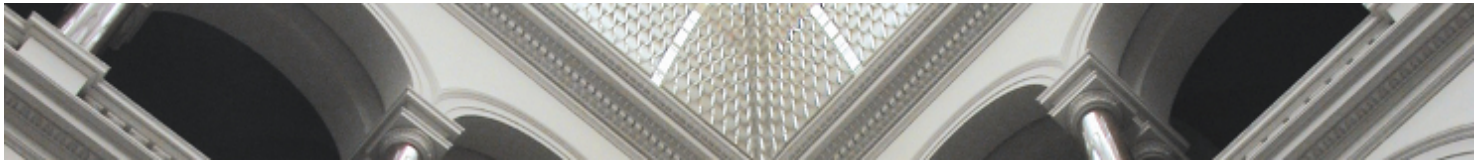
Technische Universität Berlin

Linear Regression

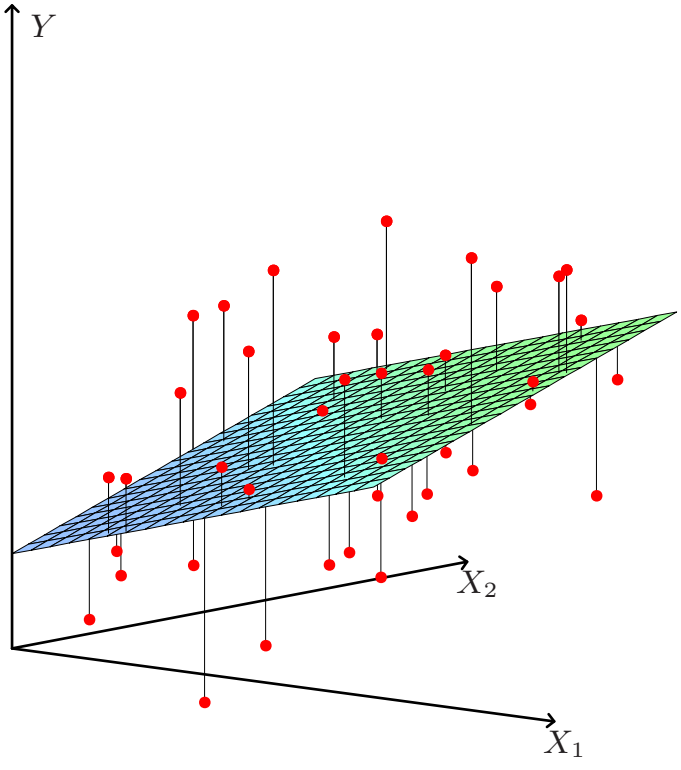


$$Y = w_1 \cdot X_1 + w_2 \cdot X_2$$





Linear Regression



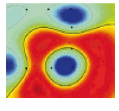
$$Y = w_1 \cdot X_1 + w_2 \cdot X_2$$

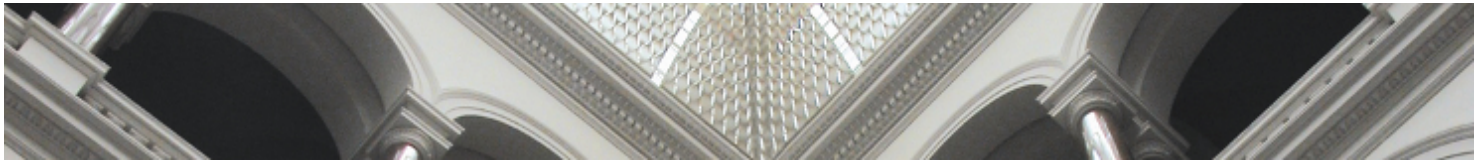
Target variable $y \in \mathbb{R}$ is modeled as a **linear combination** $\mathbf{w} \in \mathbb{R}^D$ of D features $\phi(\mathbf{x}) \in \mathbb{R}^D$

$$y = \mathbf{w}^\top \phi(\mathbf{x})$$

where $\phi(\cdot)$ denotes a set of (non-linear) functions

For the sake of simplicity we assume $\phi(\mathbf{x}) = \mathbf{x}$.





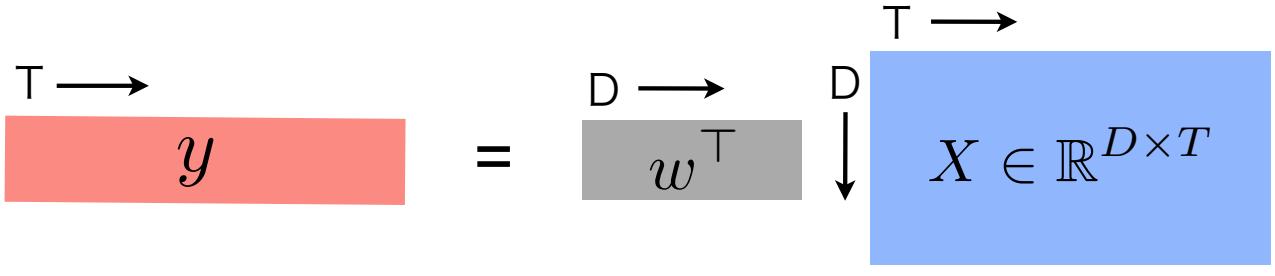
Technische Universität Berlin

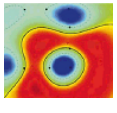
Linear Regression

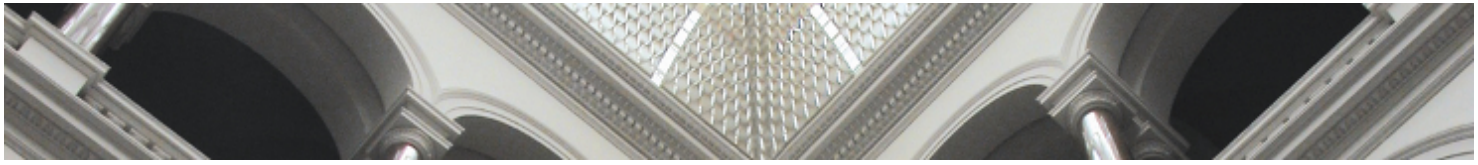
Let T be the number of samples, so $y \in \mathbb{R}^{1\times T}$ and $X \in \mathbb{R}^{D\times T}$.

The Linear Regression model in **matrix notation** then becomes

$$y = \mathbf{w}^\top X.$$







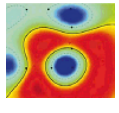
Linear Regression

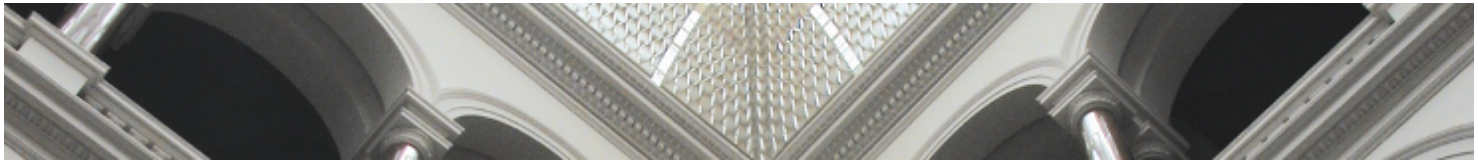
The most popular loss function to optimize **w**
is the **least-square error**

$$\mathcal{E}_{lsq}(\mathbf{w}) = \sum_{t=1}^{T} (y_t - \mathbf{w}^\top X_t)^2 \tag{2}$$

C.F. Gauß (1777-1855)

A.M. Legendre (1752-1833)



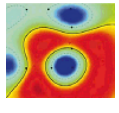


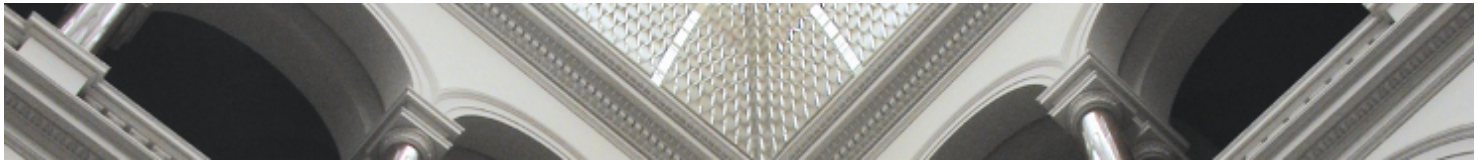
Technische Universität Berlin

Linear Regression

In matrix notation, to minimize the least-squares loss function in eq. 2

$$\begin{aligned} \mathcal{E}_{lsq}(\mathbf{w}) &= \sum_{t=1}^{T}(y_t - \mathbf{w}^\top X_t)^2 \\ &= \|y - \mathbf{w}^\top X\|^2 \\ &= yy^\top - 2\mathbf{w}^\top Xy^\top + \mathbf{w}^\top XX^\top \mathbf{w} \end{aligned}$$





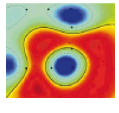
Technische Universität Berlin

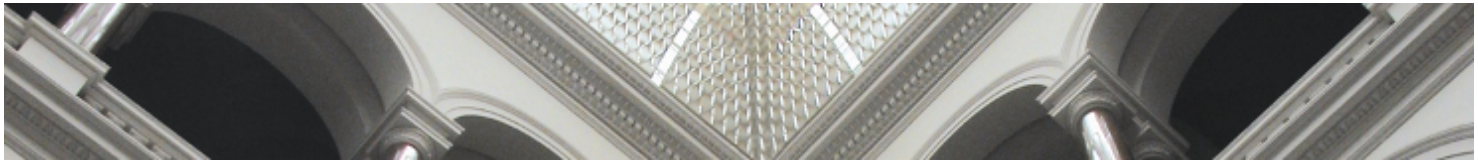
Linear Regression

In matrix notation, to minimize the least-squares loss function in eq. 2

$$\begin{aligned} \mathcal{E}_{lsq}(\mathbf{w}) &= \sum_{t=1}^{T}(y_t - \mathbf{w}^\top X_t)^2 \\ &= \|y - \mathbf{w}^\top X\|^2 \\ &= yy^\top - 2\mathbf{w}^\top Xy^\top + \mathbf{w}^\top XX^\top \mathbf{w} \end{aligned}$$

We compute derivative w.r.t. $\mathbf{w}$





Technische Universität Berlin

Linear Regression

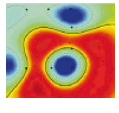
In matrix notation, to minimize the least-squares loss function in eq. 2

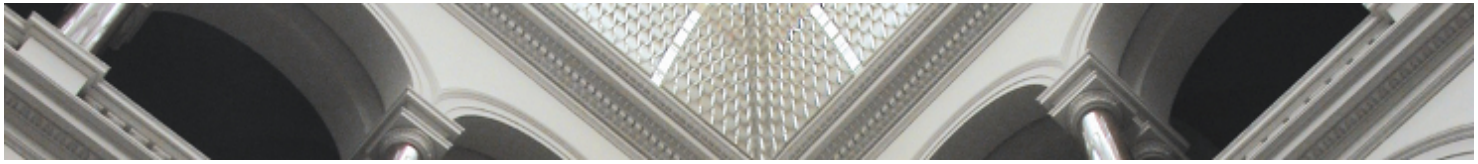
$$\begin{aligned}\mathcal{E}_{lsq}(\mathbf{w}) &= \sum_{t=1}^{T}(y_t - \mathbf{w}^\top X_t)^2 \\ &= \|y - \mathbf{w}^\top X\|^2 \\ &= yy^\top - 2\mathbf{w}^\top Xy^\top + \mathbf{w}^\top XX^\top \mathbf{w}\end{aligned}$$

We compute derivative w.r.t. $\mathbf{w}$

$$\frac{\partial \mathcal{E}_{lsq}(\mathbf{w})}{\partial \mathbf{w}} = -2Xy^\top + 2XX^\top \mathbf{w}$$

set it to zero and solve for $\mathbf{w}$





Technische Universität Berlin

Linear Regression

In matrix notation, to minimize the least-squares loss function in eq. 2

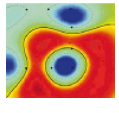
$$\begin{aligned}\mathcal{E}_{lsq}(\mathbf{w}) &= \sum_{t=1}^{T}(y_t - \mathbf{w}^\top X_t)^2 \\ &= \|y - \mathbf{w}^\top X\|^2 \\ &= yy^\top - 2\mathbf{w}^\top Xy^\top + \mathbf{w}^\top XX^\top \mathbf{w}\end{aligned}$$

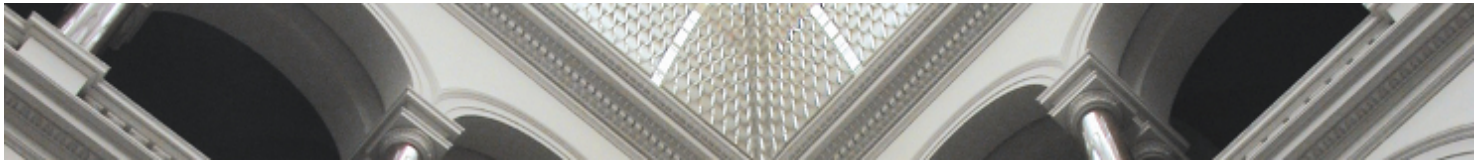
We compute derivative w.r.t. $\mathbf{w}$

$$\frac{\partial \mathcal{E}_{lsq}(\mathbf{w})}{\partial \mathbf{w}} = -2Xy^\top + 2XX^\top \mathbf{w}$$

set it to zero and solve for $\mathbf{w}$

$$\begin{aligned}-2Xy^\top + 2XX^\top \mathbf{w} &= 0 \\ XX^\top \mathbf{w} &= Xy^\top \\ \mathbf{w} &= (XX^\top)^{-1} Xy^\top\end{aligned} \tag{3}$$

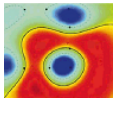


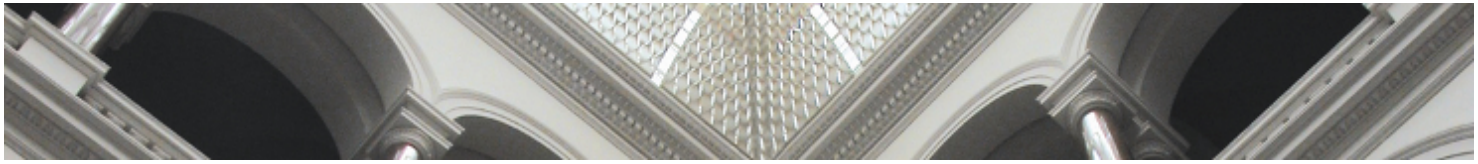


Linear Regression

For a new data point $\mathbf{z} \in \mathbb{R}^D$, we have

$$\begin{aligned} \mathbf{z} &\mapsto \mathbf{w}^T \cdot \mathbf{z} \\ \mathbf{w} &= \left(\frac{1}{T} X X^\top\right)^{-1} \frac{1}{T} X y^\top \end{aligned}$$





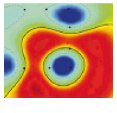
Linear Regression

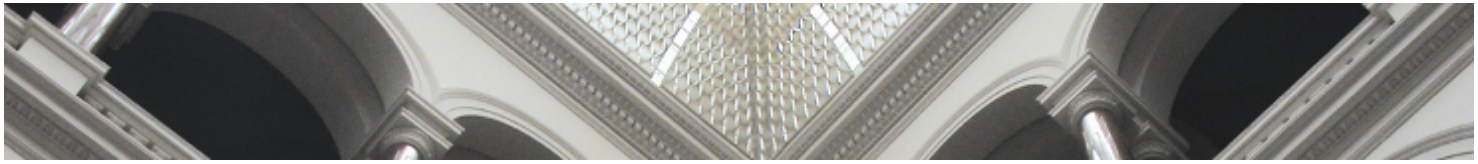
For a new data point $\mathbf{z} \in \mathbb{R}^D$, we have

$$\begin{aligned} \mathbf{z} &\mapsto \mathbf{w}^T \cdot \mathbf{z} \\ \mathbf{w} &= \left(\frac{1}{T} X X^\top\right)^{-1} \frac{1}{T} X y^\top \end{aligned}$$

Suppose our features $x^1, x^2, \ldots, x^D$ are uncorrelated and each have variance 1, $\frac{1}{T} X X^\top = I$, and features and labels have mean zero. Then:

$$\mathbf{w}^T \mathbf{z} = \frac{1}{T} y X^\top \mathbf{z}$$





Technische Universität Berlin

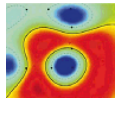
Linear Regression

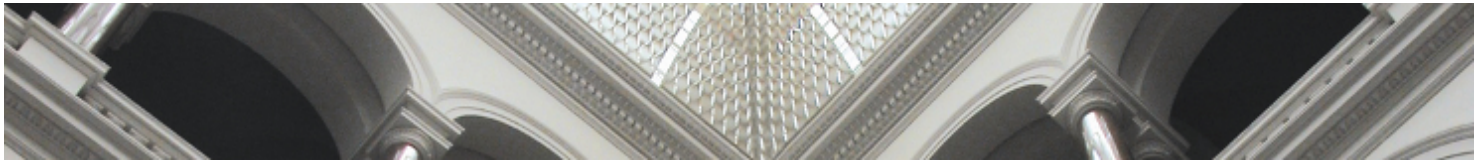
For a new data point $\mathbf{z} \in \mathbb{R}^D$, we have

$$\begin{aligned} \mathbf{z} &\mapsto \mathbf{w}^T \cdot \mathbf{z} \\ \mathbf{w} &= \left(\frac{1}{T} X X^\top\right)^{-1} \frac{1}{T} X y^\top \end{aligned}$$

Suppose our features $x^1, x^2, \ldots, x^D$ are uncorrelated and each have variance 1, $\frac{1}{T} X X^\top = I$, and features and labels have mean zero. Then:

$$\begin{aligned} \mathbf{w}^T \mathbf{z} &= \frac{1}{T} y X^\top \mathbf{z} \\ &= [Cov(x^1, y), \ldots, Cov(x^D, y)] \cdot \mathbf{z} \end{aligned}$$





Linear Regression

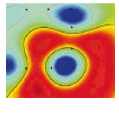
For a new data point $\mathbf{z} \in \mathbb{R}^D$, we have

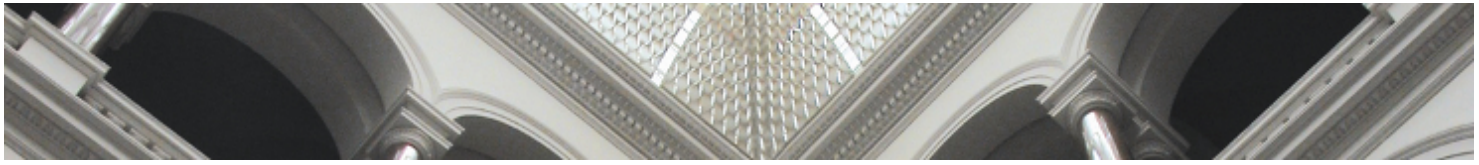
$$\begin{aligned} \mathbf{z} &\mapsto \mathbf{w}^T \cdot \mathbf{z} \\ \mathbf{w} &= \left(\frac{1}{T} X X^\top\right)^{-1} \frac{1}{T} X y^\top \end{aligned}$$

Suppose our features $x^1, x^2, \ldots, x^D$ are uncorrelated and each have variance 1, $\frac{1}{T} X X^\top = I$, and features and labels have mean zero. Then:

$$\begin{aligned} \mathbf{w}^T \mathbf{z} &= \frac{1}{T} y X^\top \mathbf{z} \\ &= [Cov(x^1, y), \ldots, Cov(x^D, y)] \cdot \mathbf{z} \\ &= Cov(x^1, y) \cdot z^1 + \ldots + Cov(x^D, y) \cdot z^D \end{aligned}$$

When the input features are uncorrelated, the estimated coefficients are equal to the univariate ones, i.e. the features do not influence each others coefficients





Technische Universität Berlin

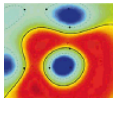
Linear Regression for Vector Labels

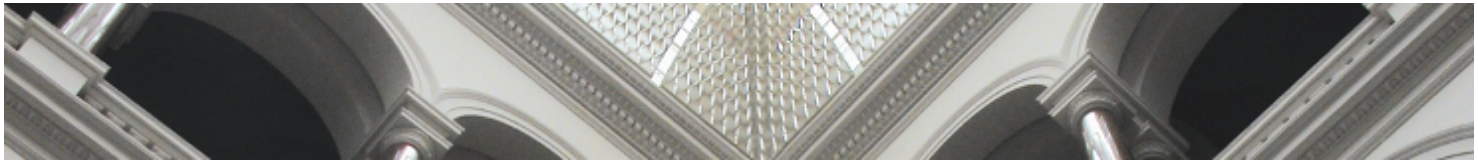
We now want to predict vector-valued labels Y

For a measurement $X \in \mathbb{R}^{D \times T}$, $Y \in \mathbb{R}^{M \times T}$ the model is

$$Y = W^\top X$$

where $W^\top \in \mathbb{R}^{M \times D}$ is a **linear mapping** from data to labels.





Linear Regression for Vector Labels

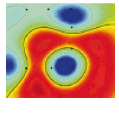
Given Data $X \in \mathbb{R}^{D \times T}$ and labels $Y \in \mathbb{R}^{M \times T}$, the error function for multiple linear regression is

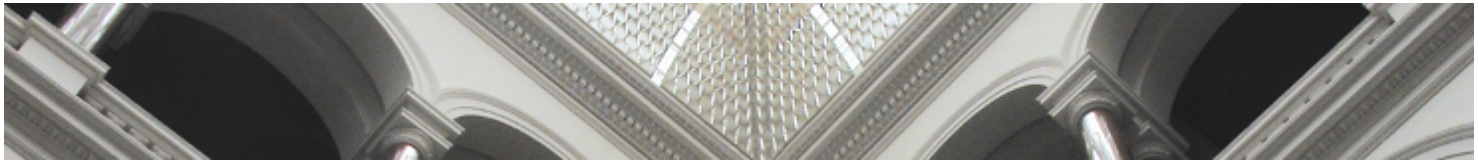
$$\mathcal{E}_{MLR}(W) = ||Y - W^\top X||_F \tag{4}$$

where $||.||_F$ denotes the Frobenius norm

Eq. 4 is minimized by (see also eq. 3)

$$W = (XX^\top)^{-1} XY^\top$$





Technische Universität Berlin

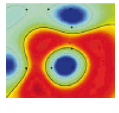
Agenda

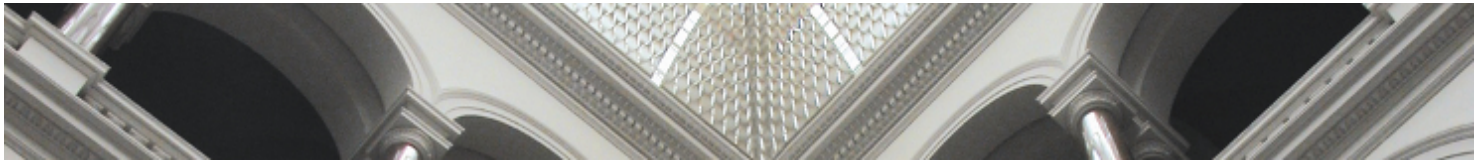
Linear Regression

Regularization & Ridge Regression

Logistic Regression

Example: Prothesis control





The statistical model of Linear Regression

Linear Model:

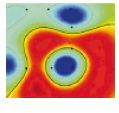
$$y = \mathbf{w}^\top \cdot X$$

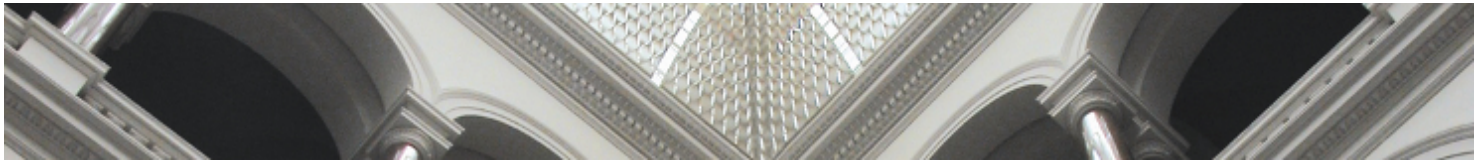
Linear Regression: estimates

$$\hat{\mathbf{w}} = (XX^\top)^{-1}Xy$$

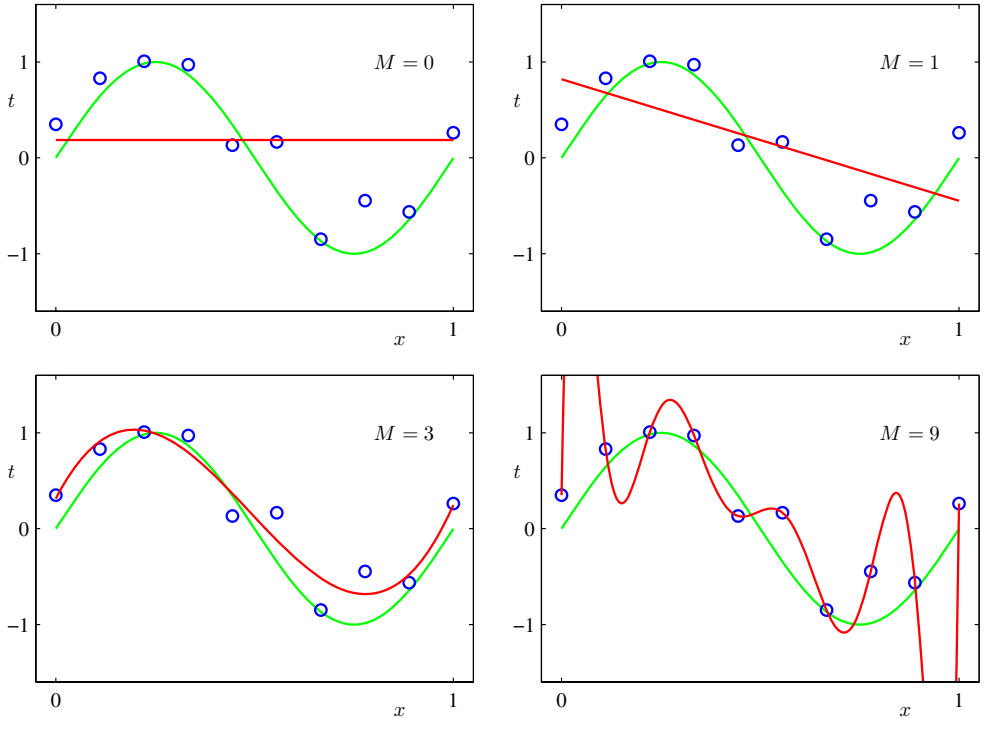
from given data X, y.

$\hat{\mathbf{w}}$ is a function of the data and thus itself a random variable

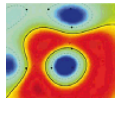


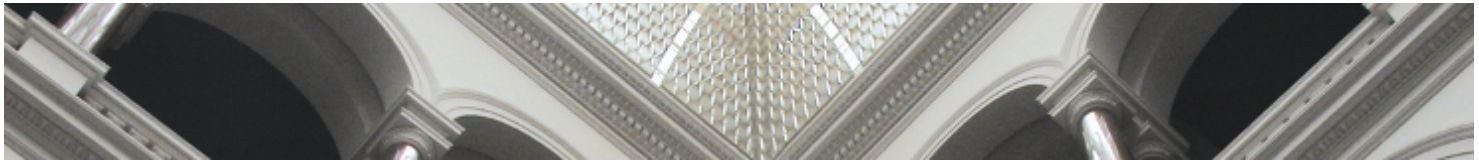


Polynomial regression



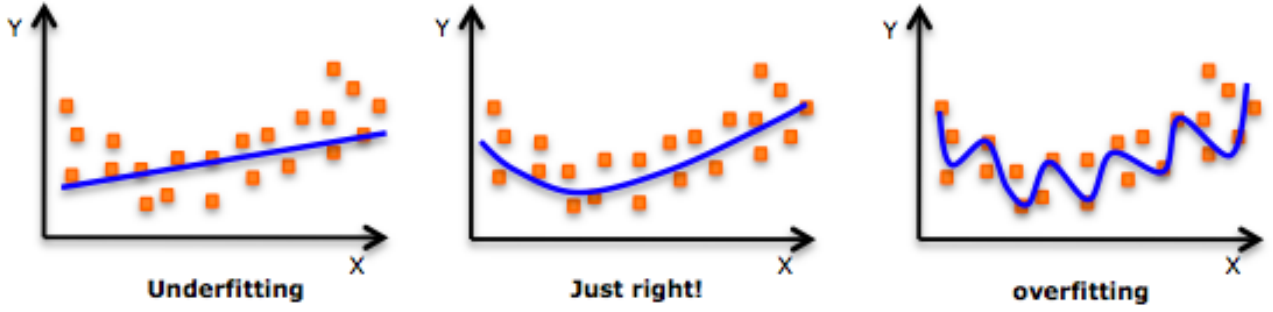
$$f(x) = w_0 + w_1 \cdot x^1 + \ldots + w_M \cdot x^M$$

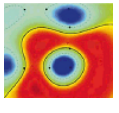


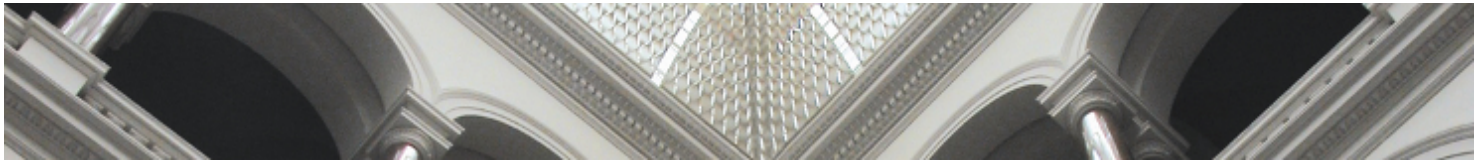


Technische Universität Berlin

Over-fitting vs. Under-fitting

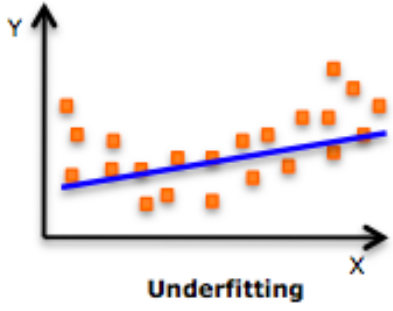


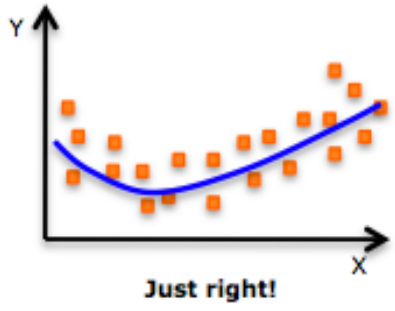


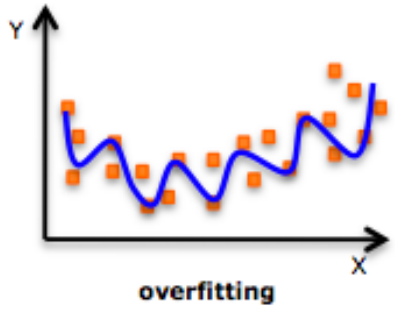


Technische Universität Berlin

Over-fitting vs. Under-fitting

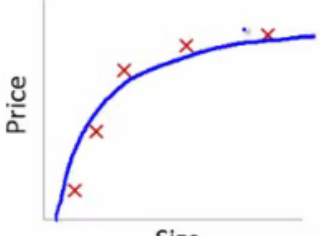






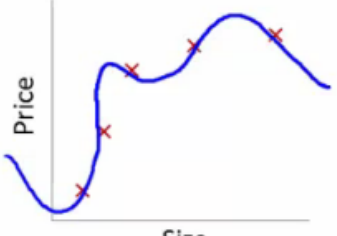
$\theta_0 + \theta_1 x$

High bias (underfit)



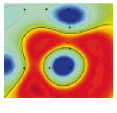
$\theta_0 + \theta_1 x + \theta_2 x^2$

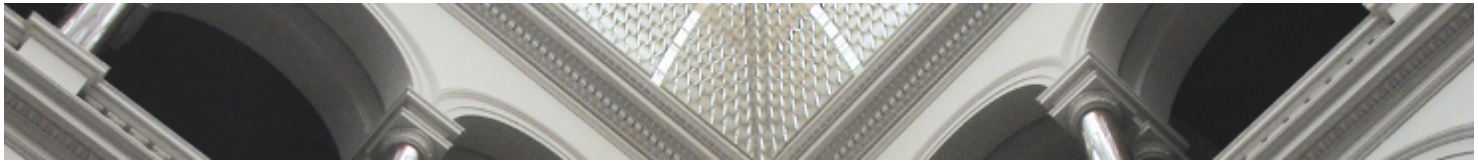
"Just right"



$\theta_0 + \theta_1 x + \theta_2 x^2 + \theta_3 x^3 + \theta_4 x^4$

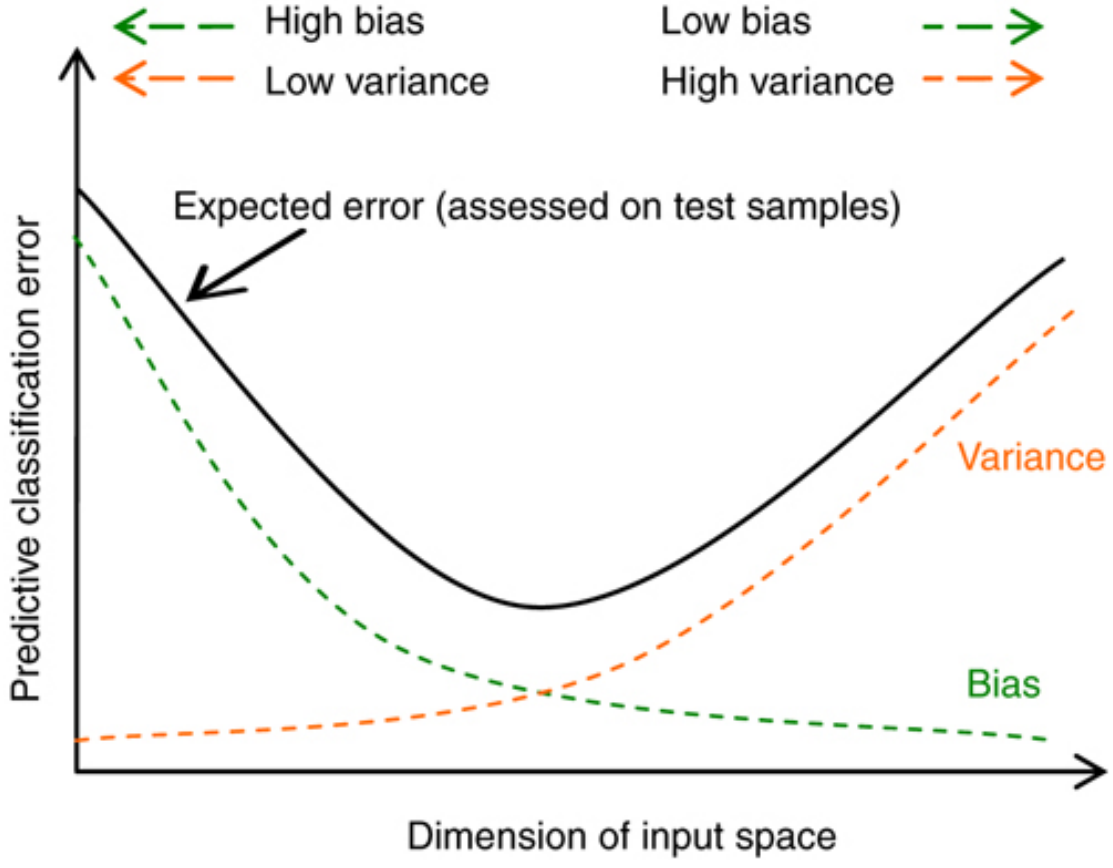
High variance (overfit)

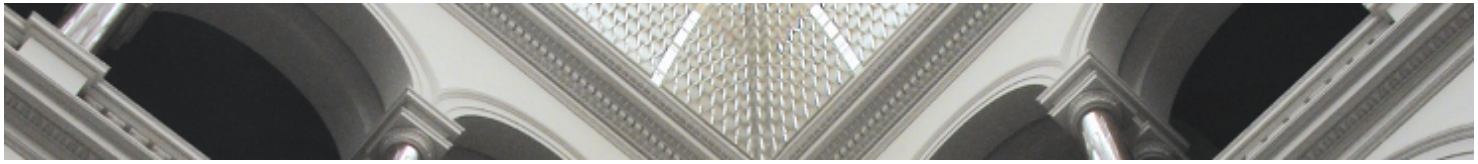




Technische Universität Berlin

The Bias-Variance Tradeoff



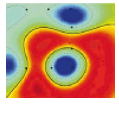


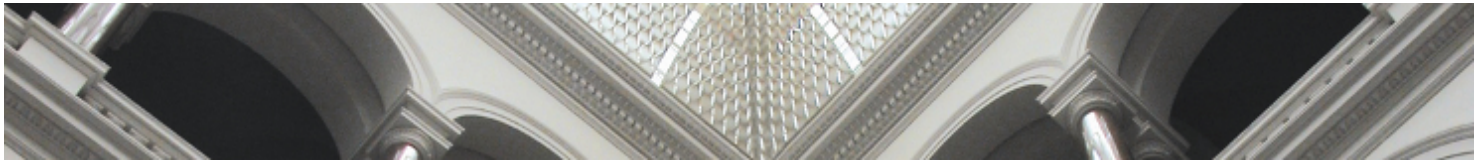
Ridge Regression

Often it is important to **control the complexity** of the solution **w**.

Regularization: This is done by constraining the norm of **w**,

$$\mathcal{E}_{RR}(\mathbf{w}) = ||y - \mathbf{w}^\top X||^2 + \lambda ||\mathbf{w}||^2$$





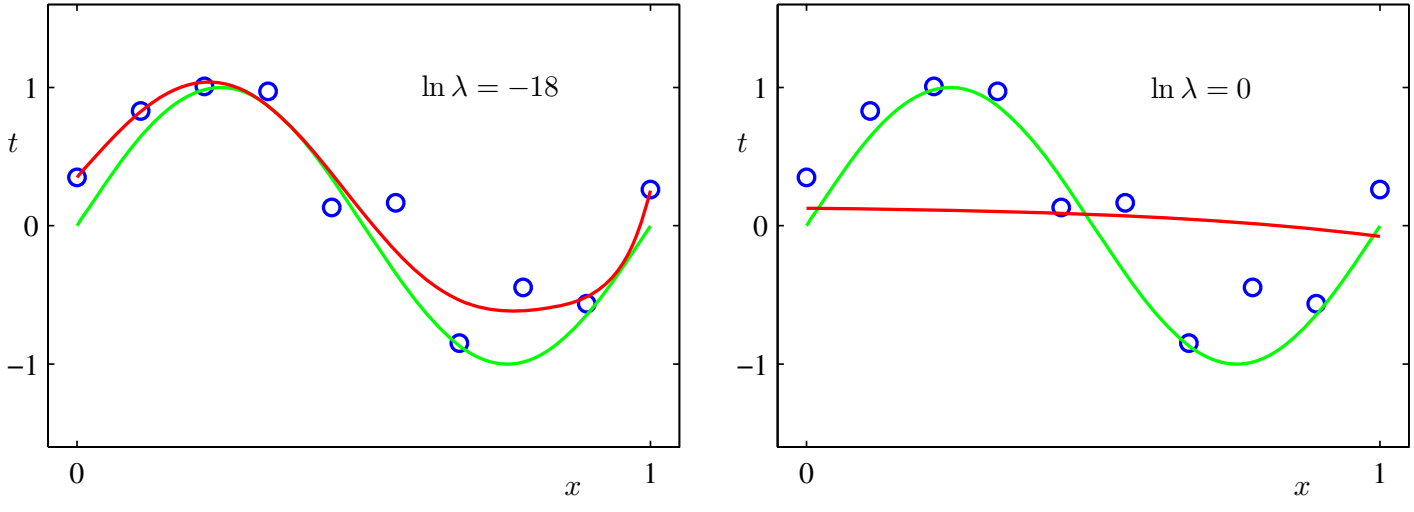
Technische Universität Berlin

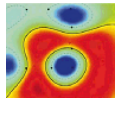
Ridge Regression

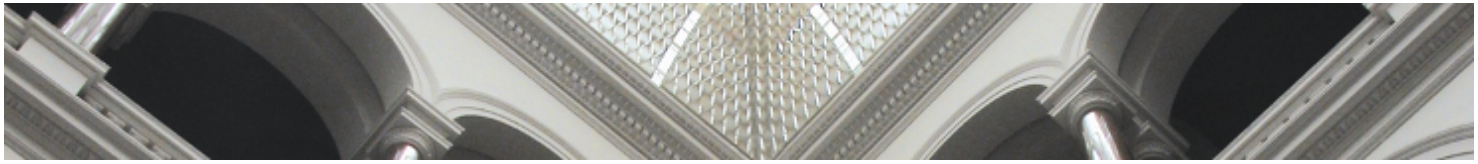
Often it is important to **control the complexity** of the solution **w**.

Regularization: This is done by constraining the norm of **w**,

$$\mathcal{E}_{RR}(\mathbf{w}) = ||y - \mathbf{w}^\top X||^2 + \lambda ||\mathbf{w}||^2$$







Ridge Regression

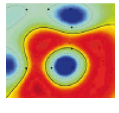
Computing the derivative w.r.t. w yields

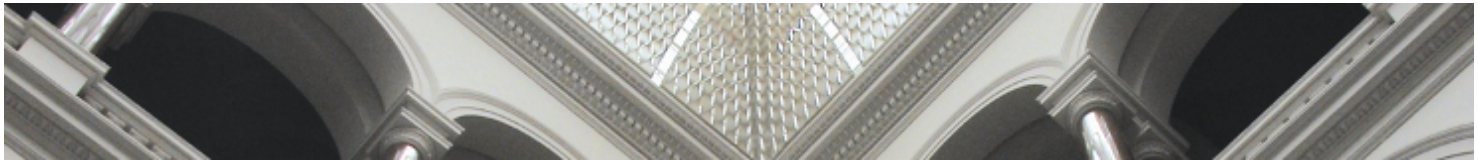
$$\frac{\partial \mathcal{E}_{RR}(\mathbf{w})}{\partial \mathbf{w}} = -2Xy^\top + 2XX^\top \mathbf{w} + \lambda 2\mathbf{w}.$$

Setting the gradient to zero and rearranging terms the optimal $\mathbf{w}$ is

$$\begin{aligned} 2XX^\top \mathbf{w} + \lambda 2\mathbf{w} &= 2Xy^\top \\ (XX^\top + \lambda I)\mathbf{w} &= Xy^\top \\ \mathbf{w} &= (XX^\top + \lambda I)^{-1} Xy^\top \end{aligned}$$

$\Rightarrow$ Biased estimator, but smaller variance





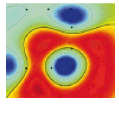
Lasso Regression

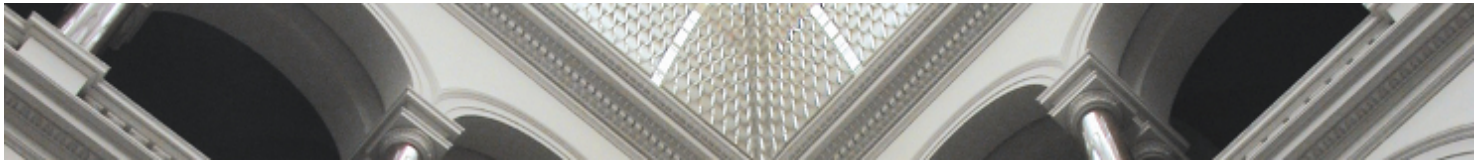
We saw that ridge regression

$$\mathcal{E}_{RR}(\mathbf{w}) = ||y - \mathbf{w}^\top X||^2 + \lambda ||\mathbf{w}||^2$$

can have better prediction error than linear regression in a variety of scenarios, depending on the choice of λ. It worked best when there was a subset of the true coefficients that are small or zero.

But it will never sets coefficients to zero exactly, and therefore cannot perform **variable selection** in the linear model. While this didn't seem to hurt its prediction ability, it is not desirable for the purposes of interpretation (especially if the number of variables p is large).





Lasso Regression

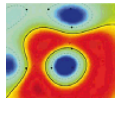
The **lasso** objective is defined as

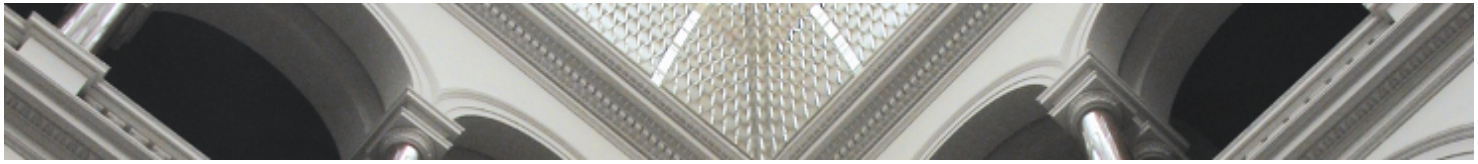
$$\mathcal{E}_{lasso}(\mathbf{w}) = ||y - \mathbf{w}^\top X||^2 + \lambda ||\mathbf{w}||_1$$

$$\mathcal{E}_{lasso}(\mathbf{w}) = ||y - \mathbf{w}^\top X||^2 + \lambda \sum_{i=1}^{D} |w_i|$$

The only difference between the lasso problem and ridge regression is that the latter uses a (squared) ℓ_2 penalty $||\mathbf{w}||_2^2$, while the former uses an ℓ_1 penalty $||\mathbf{w}||_1$. But even though these problems look similar, their solutions behave very differently.

Note the name “lasso” is actually an acronym for: Least Absolute Selection and Shrinkage Operator.



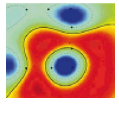


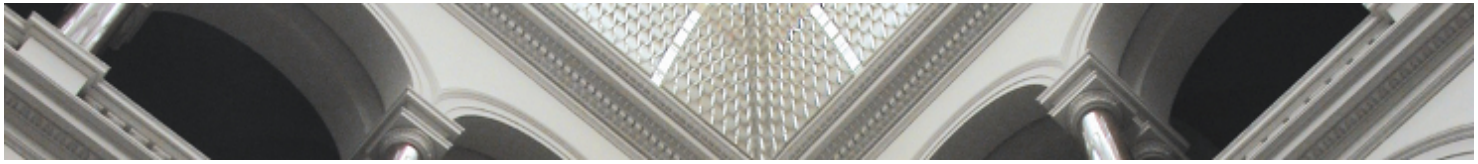
Technische Universität Berlin

Lasso Regression

The tuning parameter λ controls the strength of the penalty, and (like ridge regression) we get $\hat{\mathbf{w}}_{lasso}$ = the linear regression estimate when $\lambda = 0$, and $\hat{\mathbf{w}}_{lasso} = 0$ when $\lambda = \infty$.

For λ in between these two extremes, we are balancing two ideas: fitting a linear model of y on X, and shrinking the coefficients. But the nature of the ℓ_1 penalty causes some coefficients to be shrunken to zero exactly.



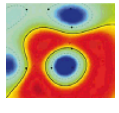


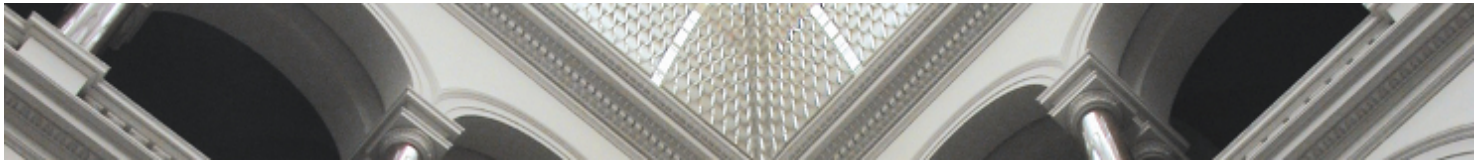
Lasso Regression

The tuning parameter λ controls the strength of the penalty, and (like ridge regression) we get $\hat{\mathbf{w}}_{lasso}$ = the linear regression estimate when $\lambda = 0$, and $\hat{\mathbf{w}}_{lasso} = 0$ when $\lambda = \infty$.

For λ in between these two extremes, we are balancing two ideas: fitting a linear model of y on X, and shrinking the coefficients. But the nature of the ℓ_1 penalty causes some coefficients to be shrunken to zero exactly.

Generally speaking:
- The bias increases as λ (amount of shrinkage) increases.
- The variance decreases as λ (amount of shrinkage) increases.





Technische Universität Berlin

Regularization

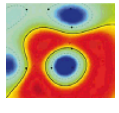
It can be helpful to think of our two problems **constrained form**:

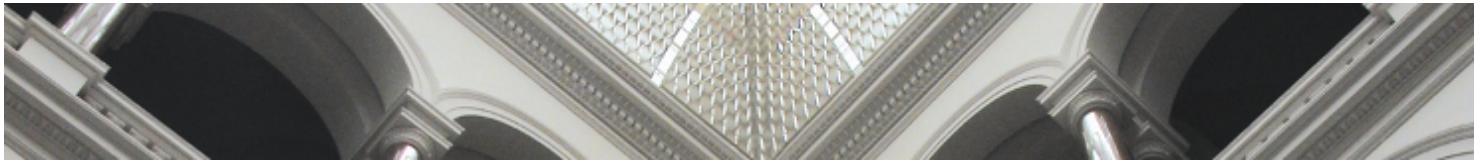
$$\hat{\mathbf{w}}_{ridge} = \arg\min_{\mathbf{w}} ||y - \mathbf{w}^\top X||^2 \quad \text{subject to } ||\mathbf{w}||_2^2 \leq t$$

$$\hat{\mathbf{w}}_{lasso} = \arg\min_{\mathbf{w}} ||y - \mathbf{w}^\top X||^2 \quad \text{subject to } ||\mathbf{w}||_1 \leq t$$

Now t is the tuning parameter (before it was λ). For any λ and corresponding solution in the previous formulation (sometimes called penalized form), there is a value of t such that the above constrained form has this same solution.

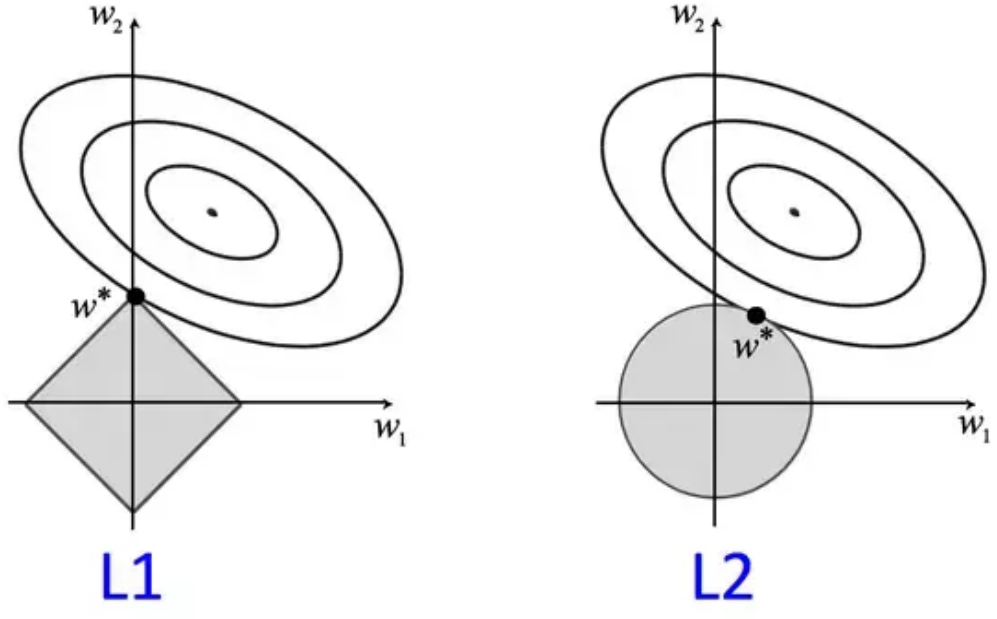
In comparison, the usual linear regression estimate solves the **unconstrained** least squares problem; these estimates constrain the coefficient vector to lie in some geometric shape centered around the origin. This generally reduces the variance because it keeps the estimate close to zero. But which shape we choose really matters!



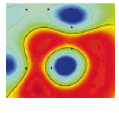


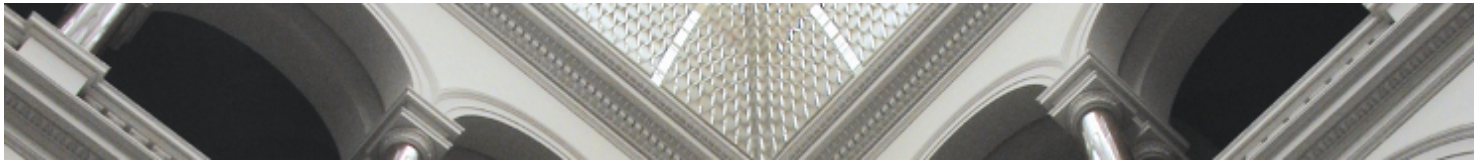
Regularization

ℓ_1 regularization leads to sparsity



Link: Regularization video





Technische Universität Berlin

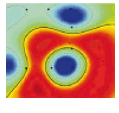
Agenda

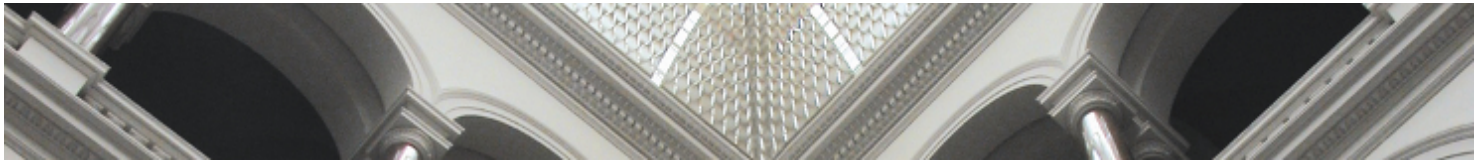
Linear Regression

Regularization & Ridge Regression

Logistic Regression

Example: Prothesis control





Technische Universität Berlin

Regression (Recap)

x_1 (hours)	x_2 (attendance)	y (score)
10	5	90
9	5	80
3	2	50
2	4	60
11	1	40

Hypothesis:

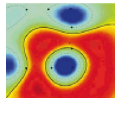
$$Y = W^\top X$$

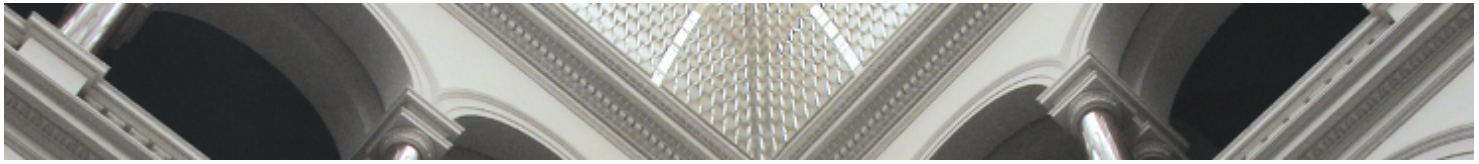
Cost:

$$\mathcal{E}(W) = \frac{1}{T}\sum_{t=1}^{T}(W^T X - Y)^2$$

In order to minimize **least-square error** to find the ”best fit ”, compute derivative:

$$\frac{\partial \mathcal{E}(W)}{\partial W}$$

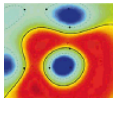


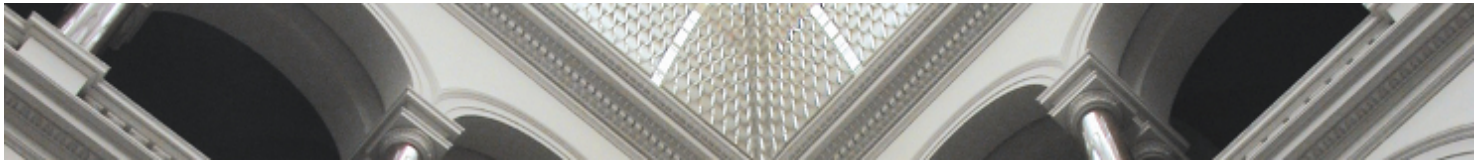


Technische Universität Berlin

(Binary) Classification

1. Spam detection: Spam or Ham
2. Facebook feed: show or hide
3. Credit card fraudulent transaction detection: legitimate/fraud

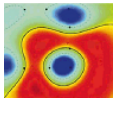


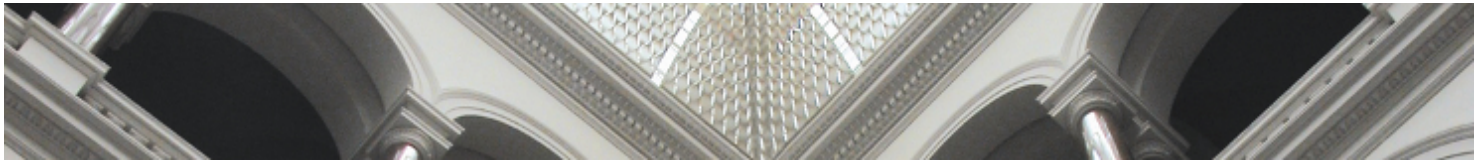


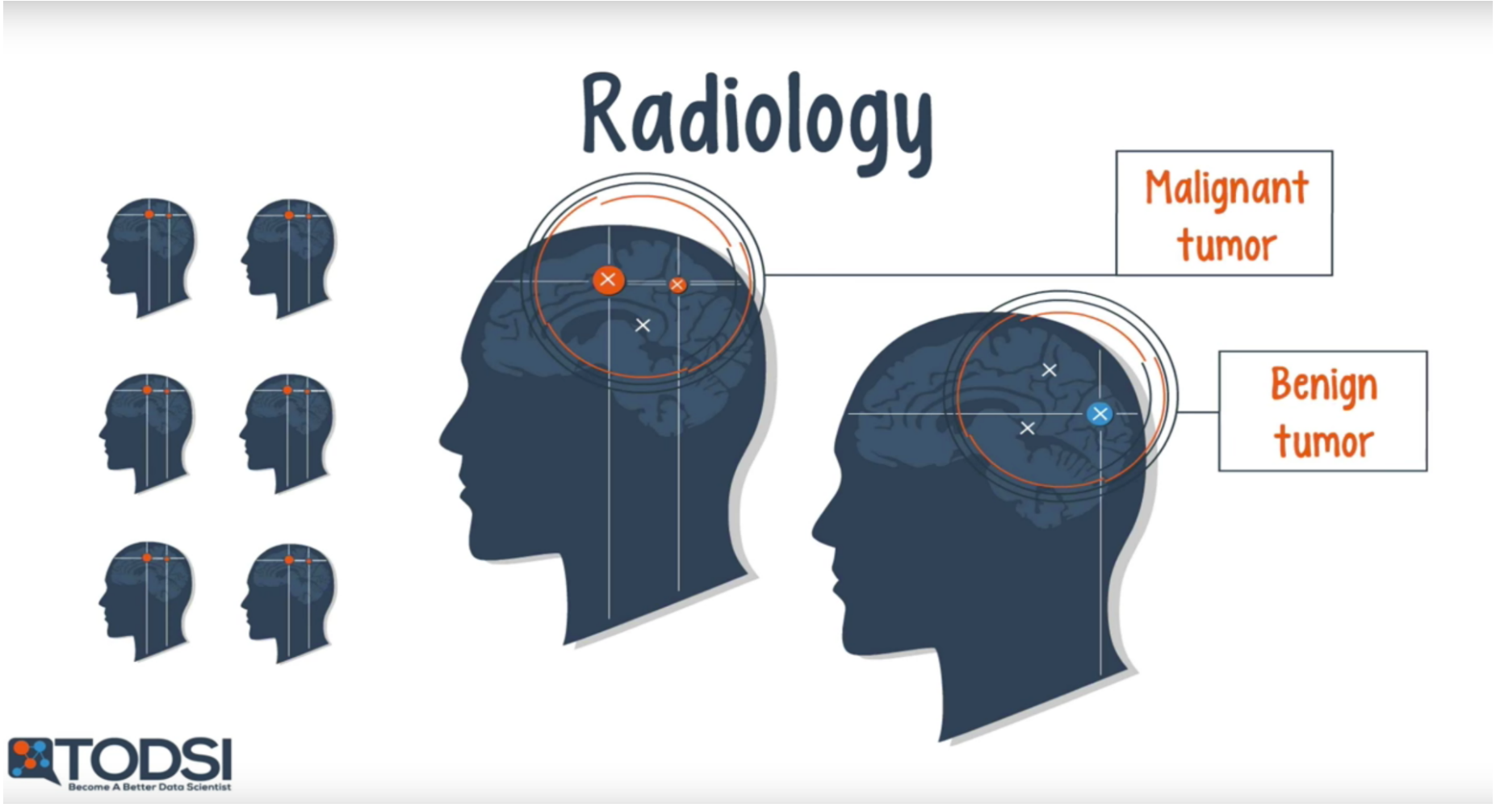
Technische Universität Berlin

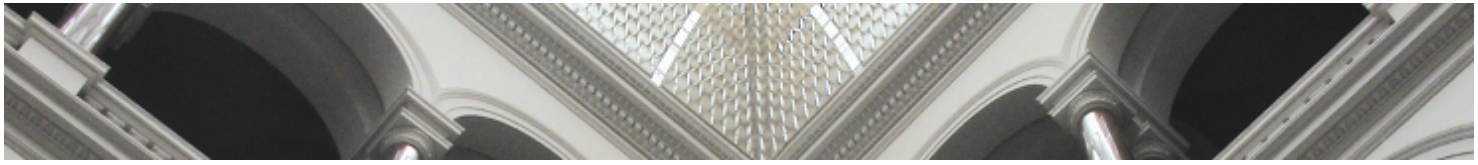
0, 1 encoding

1. Spam detection: Spam (1) or Ham (0)
2. Facebook feed: show (1) or hide (0)
3. Credit card fraudulent transaction detection: legitimate (0) or fraud (1)



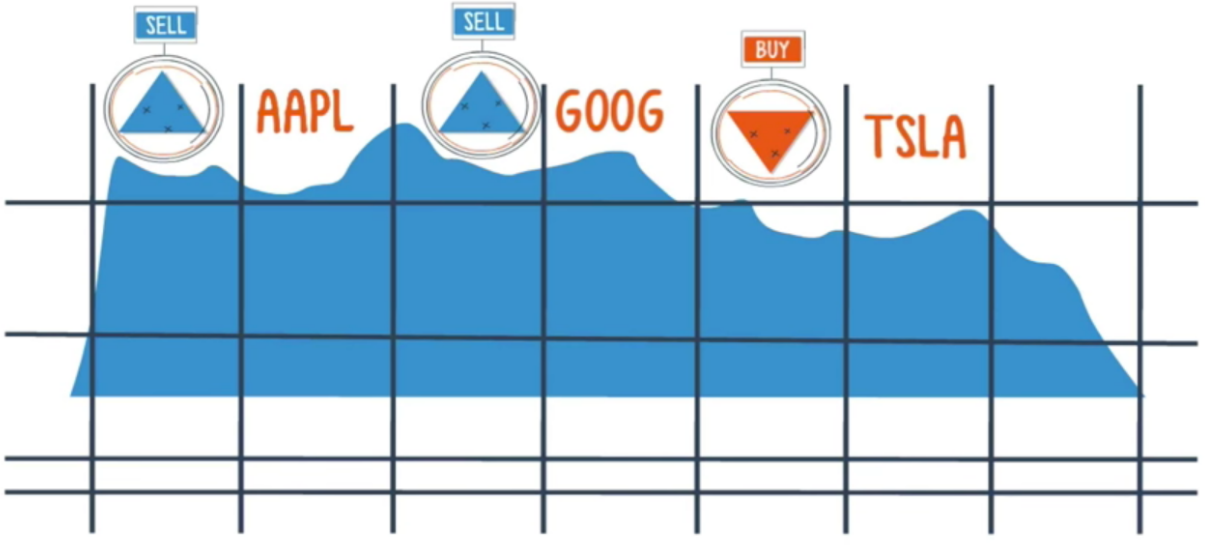



Radiology
Malignant tumor
Benign tumor
TODSI
Become A Better Data Scientist

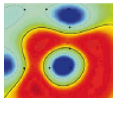


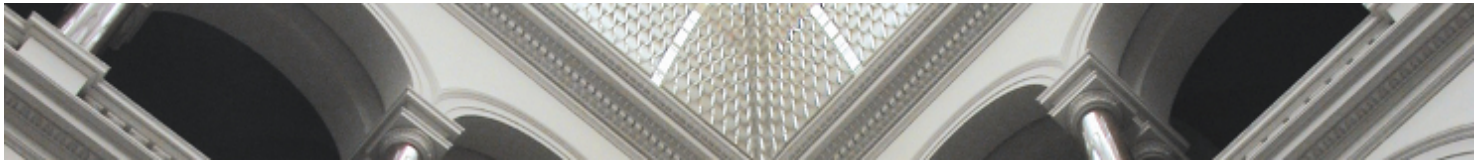
Finance

DWJI	17,499.10	▼
SP500	2,025.51	▼
NASDAQ	4,976.9	▲
AAPL	107.71	▲
GOOG	750.06	▲
TSLA	234.24	▼



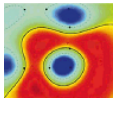
TODSI
Become A Better Data Scientist

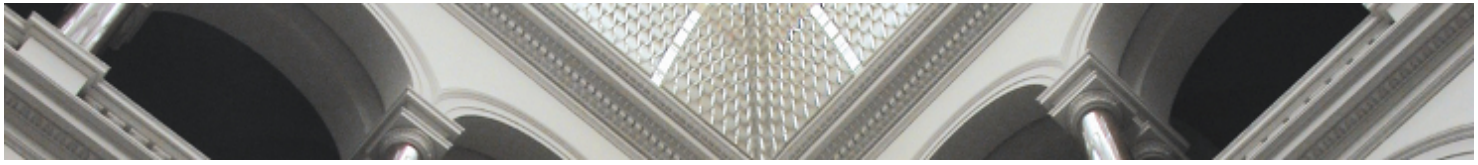




Pass(1) / Fail(0) based on study hours

In the previous table, it seems clear that study hours have an effect on someone's exam score, how do we come up with a model that will let us explore this relationship?

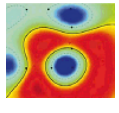


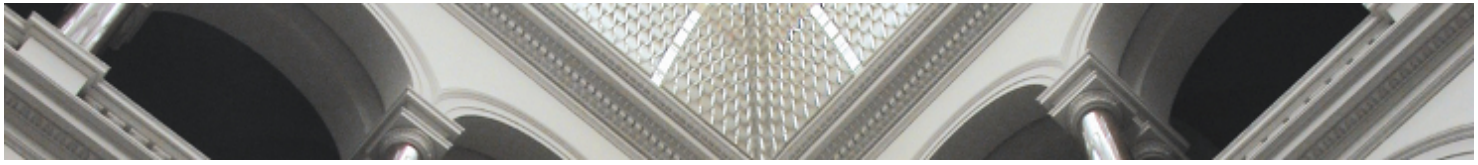


Pass(1) / Fail(0) based on study hours

In the previous table, it seems clear that study hours have an effect on someone's exam score, how do we come up with a model that will let us explore this relationship?

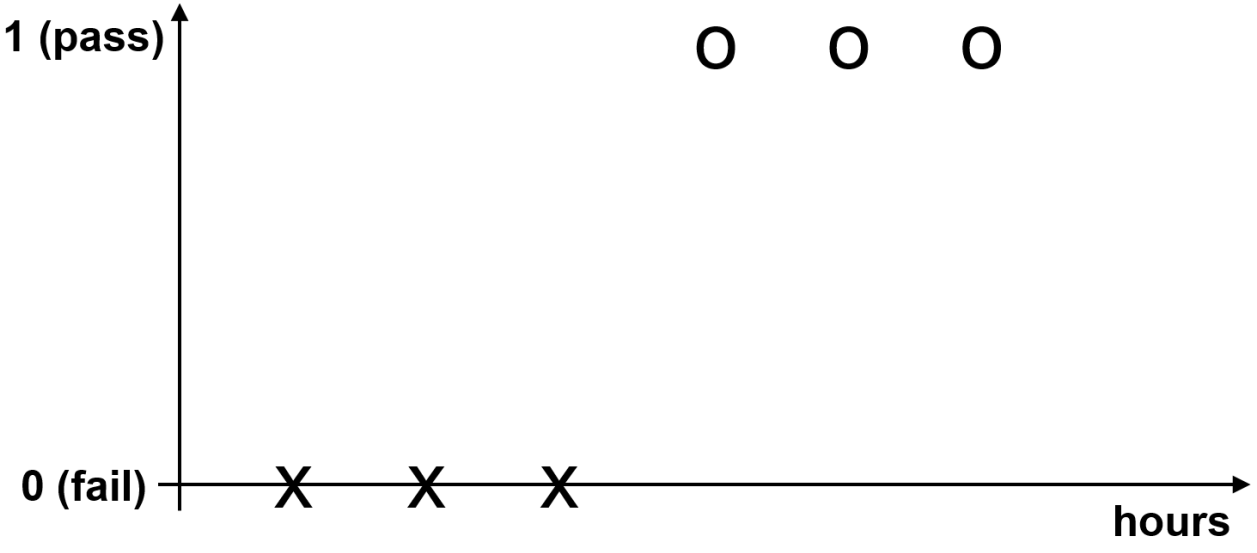
One way to think about the problem - we can treat Pass (1) and Fail (0) as successes and failures given by a transformation of a linear model.

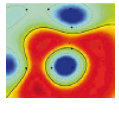


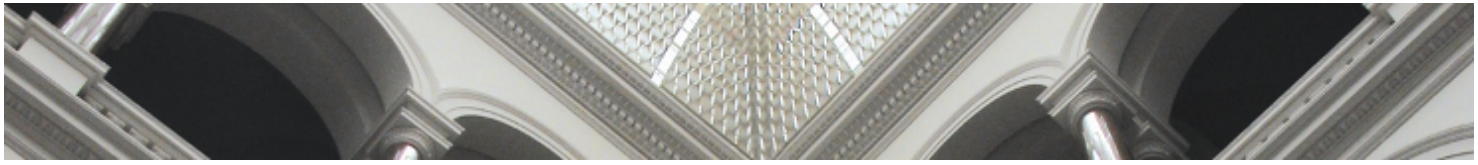


Technische Universität Berlin

Linear Regression?

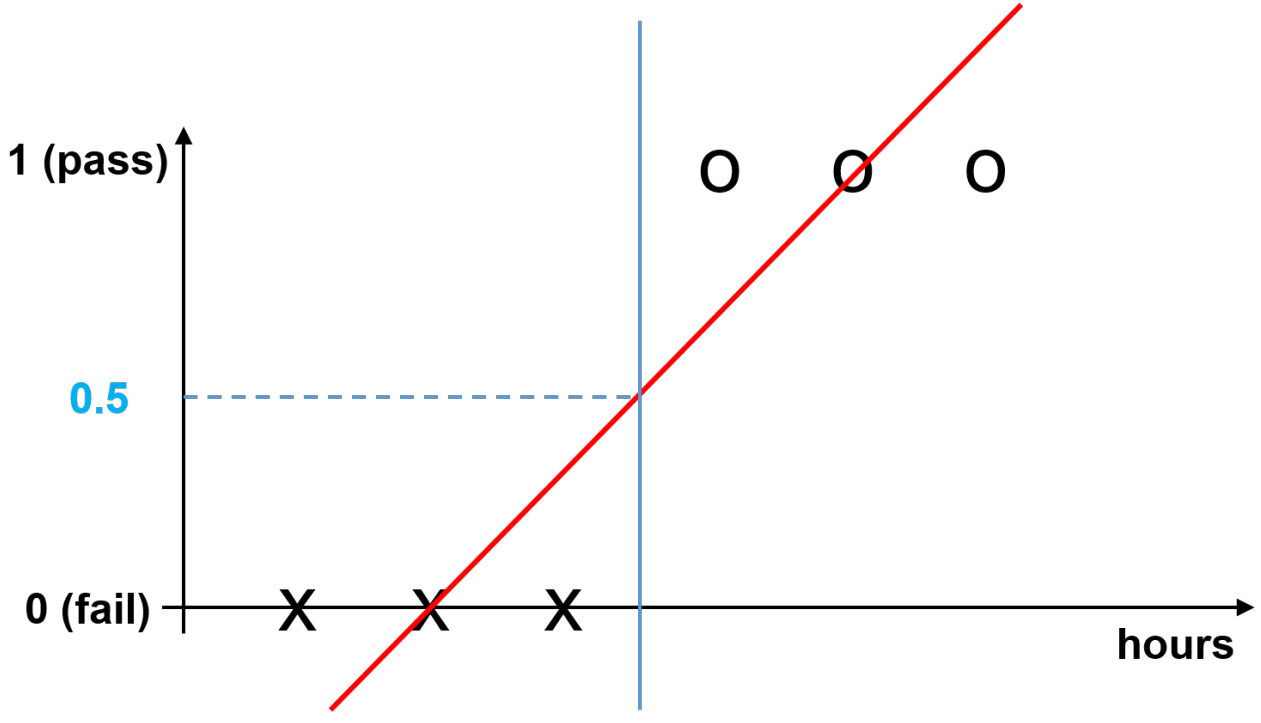


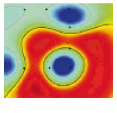


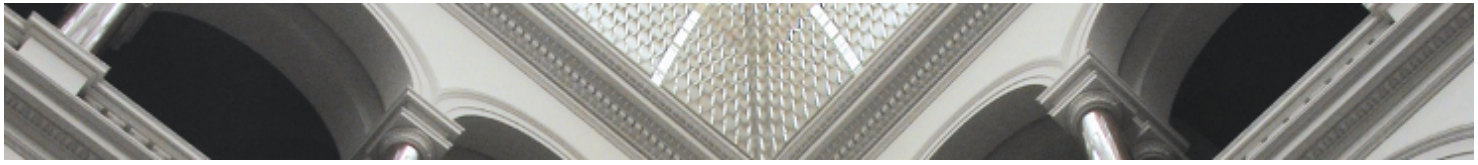


Technische Universität Berlin

Linear Regression?

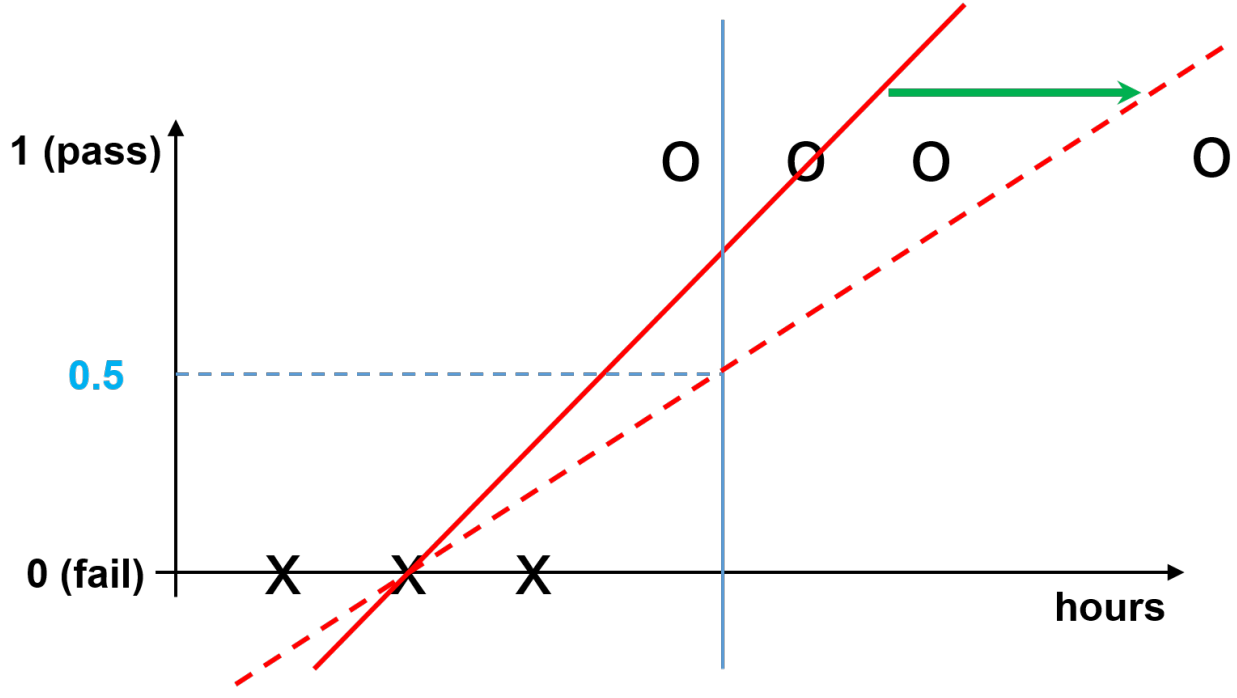






Technische Universität Berlin

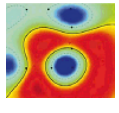
Linear Regression?



We know Y is 0 or 1

$$f(X) = W^\top X$$

Hypothesis can give values large than 1 or less than 0

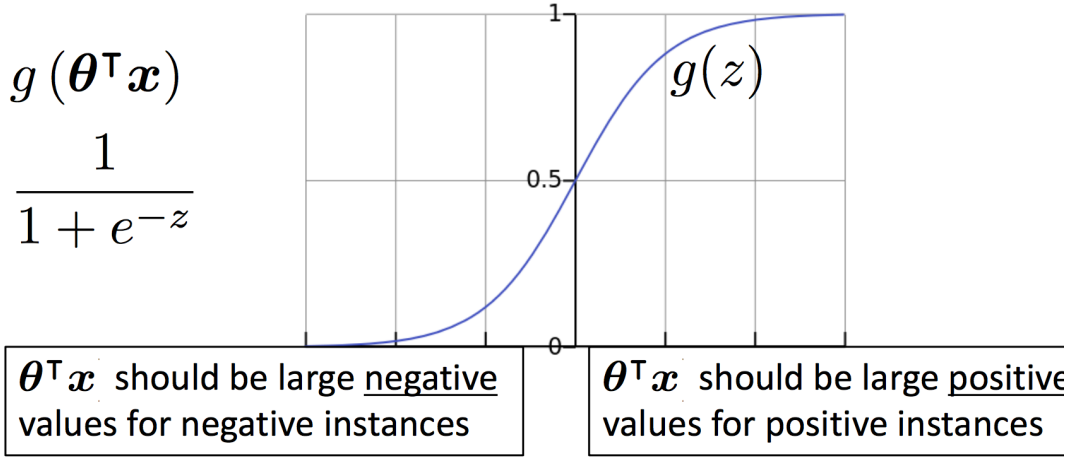


Logistic Regression

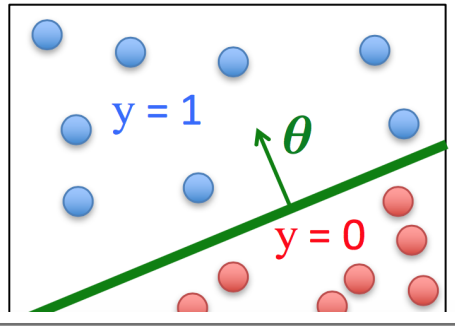
The logistic (= sigmoid) function takes a value between 0 and 1 and maps it to a value between $-\infty$ and ∞.

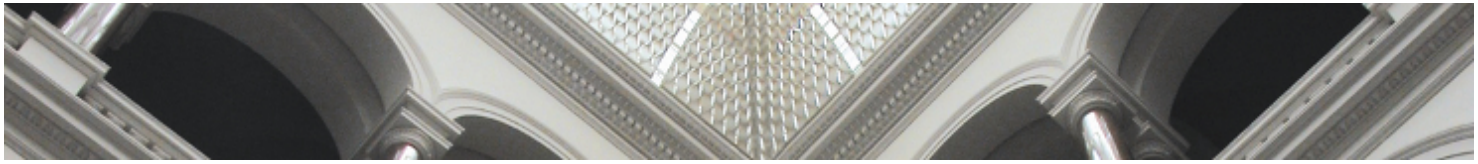
$$h_{\boldsymbol{\theta}}(\boldsymbol{x}) = g\left(\boldsymbol{\theta}^{\mathsf{T}}\boldsymbol{x}\right)$$

$$g(z) = \frac{1}{1+e^{-z}}$$



- Assume a threshold and...
  - Predict y = 1 if $h_{\boldsymbol{\theta}}(\boldsymbol{x}) \geq 0.5$
  - Predict y = 0 if $h_{\boldsymbol{\theta}}(\boldsymbol{x}) < 0.5$





Technische Universität Berlin

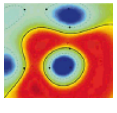
Agenda

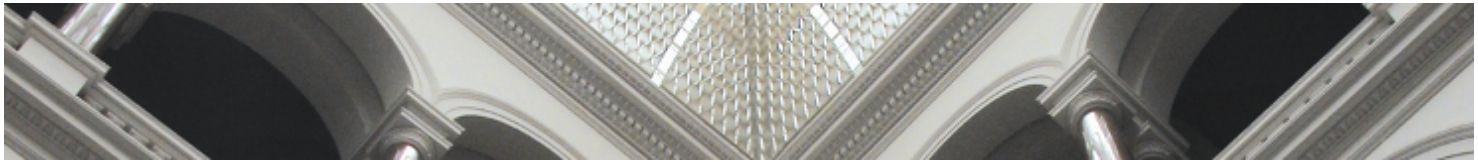
Linear Regression

Regularization & Ridge Regression

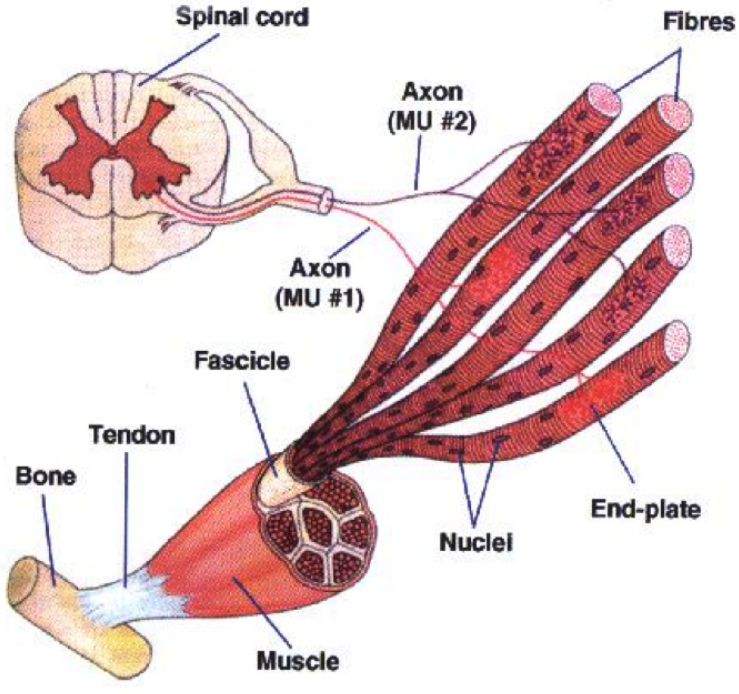
Logistic Regression

Example: Prothesis control



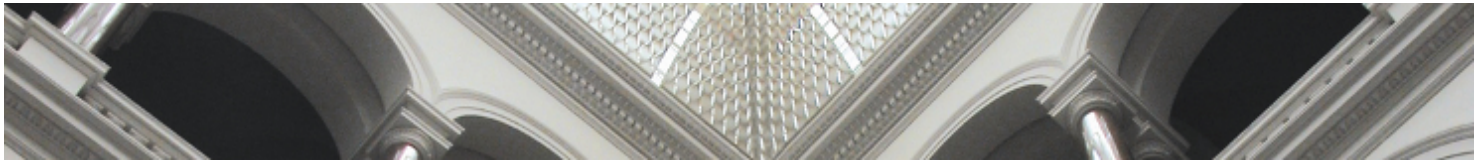


Application Example:
Myoelectric Control of Prostheses

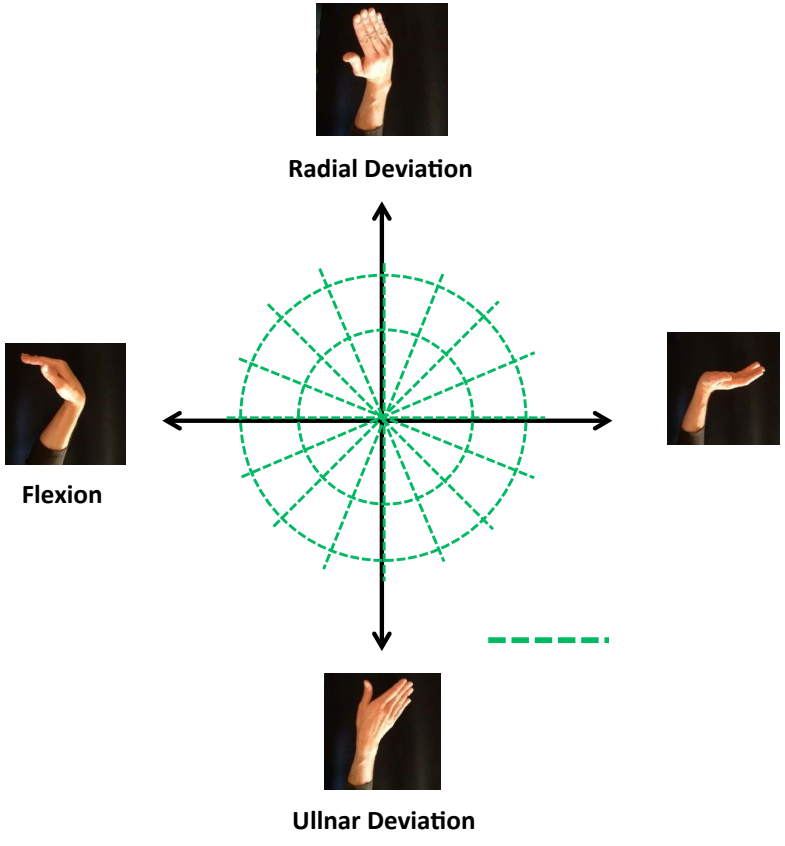


Neurons activate muscles via electric discharges
Electric activity can be measured non-invasively

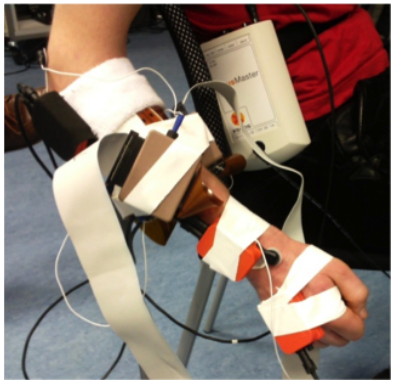
State-of-the-art hand prosthesis
Only 2 degrees of freedom are controlled (open/close, rotate)
Controlled by muscle activity



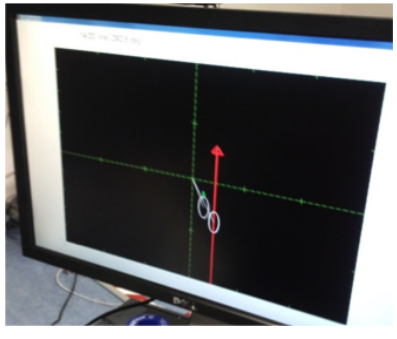
Acquisition of Training Data



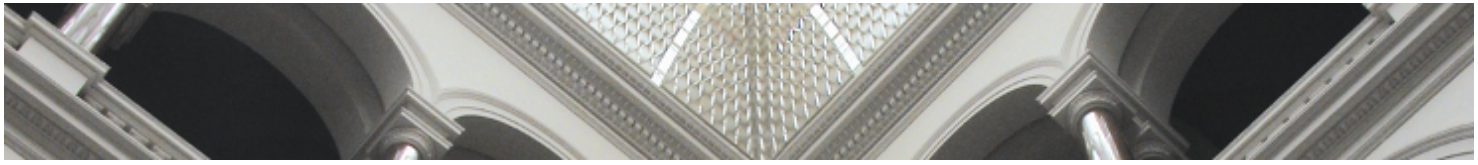
Experimental Paradigm



Motion Capture System

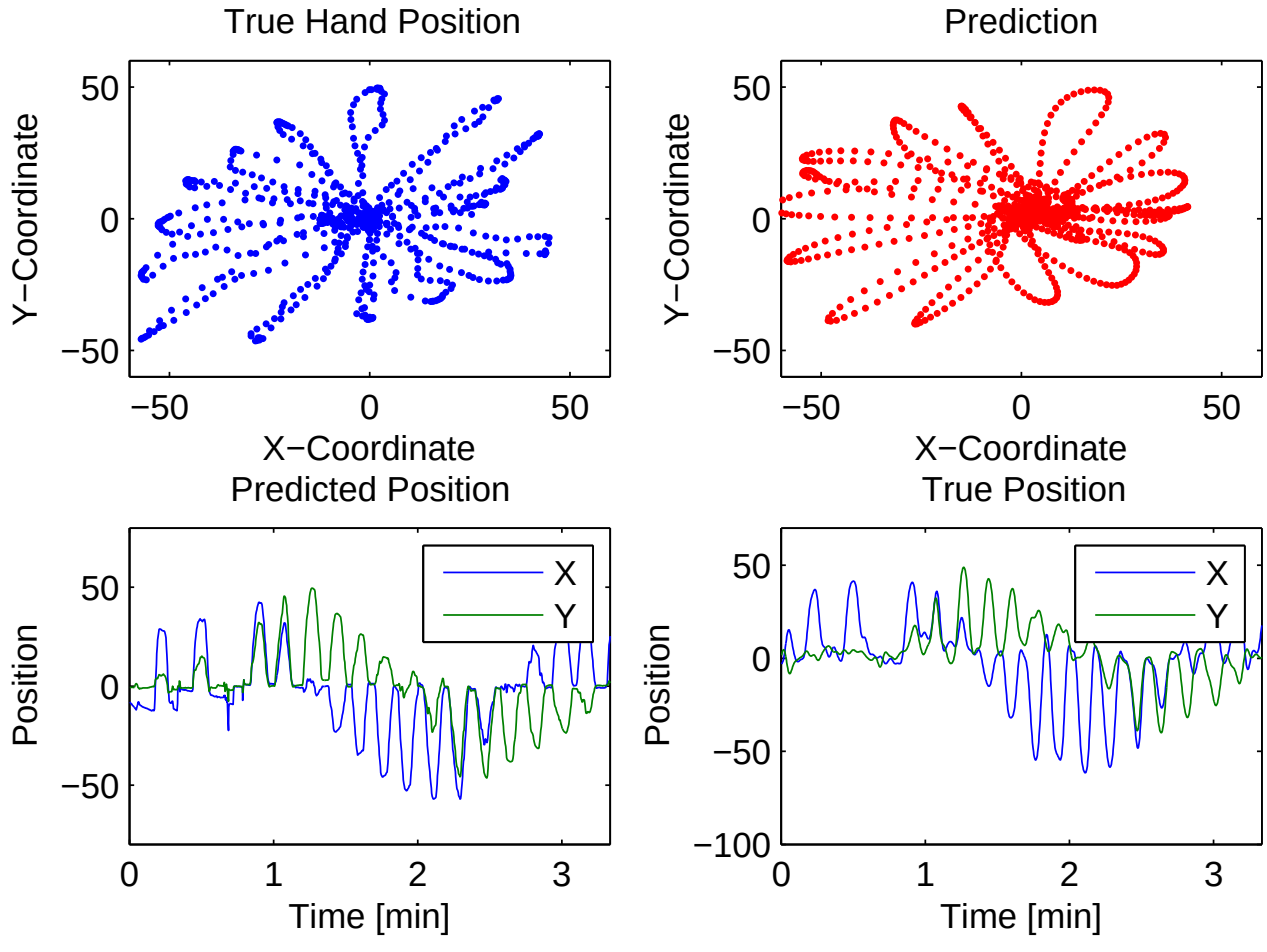


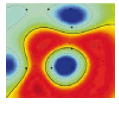
Visual Feedback

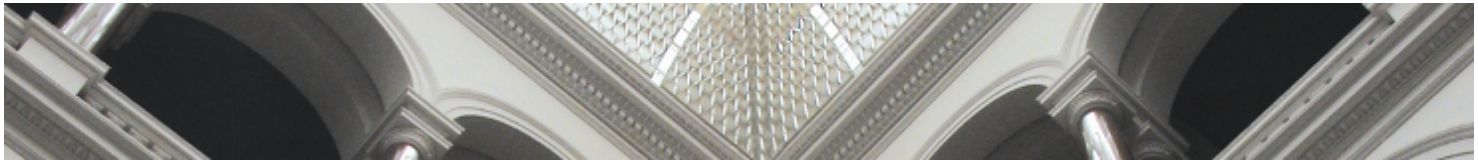


Technische Universität Berlin

Results Linear Regression - Smoothed







Technische Universität Berlin

Thank you!