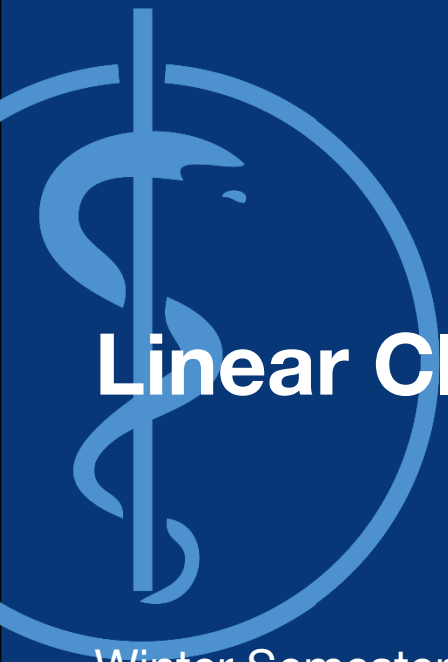


Linear Classifier

Winter Semester 2016

Dr. Tingying Peng

Technische Universität München

JOHNS HOPKINS
WHITING SCHOOL
of ENGINEERING

Table of Content

- General Information of Linear Model
- Ordinary Linear Regression
- Optimal Separating Hyperplanes
- Evaluation Measure
- Logistic Regression

General introduction

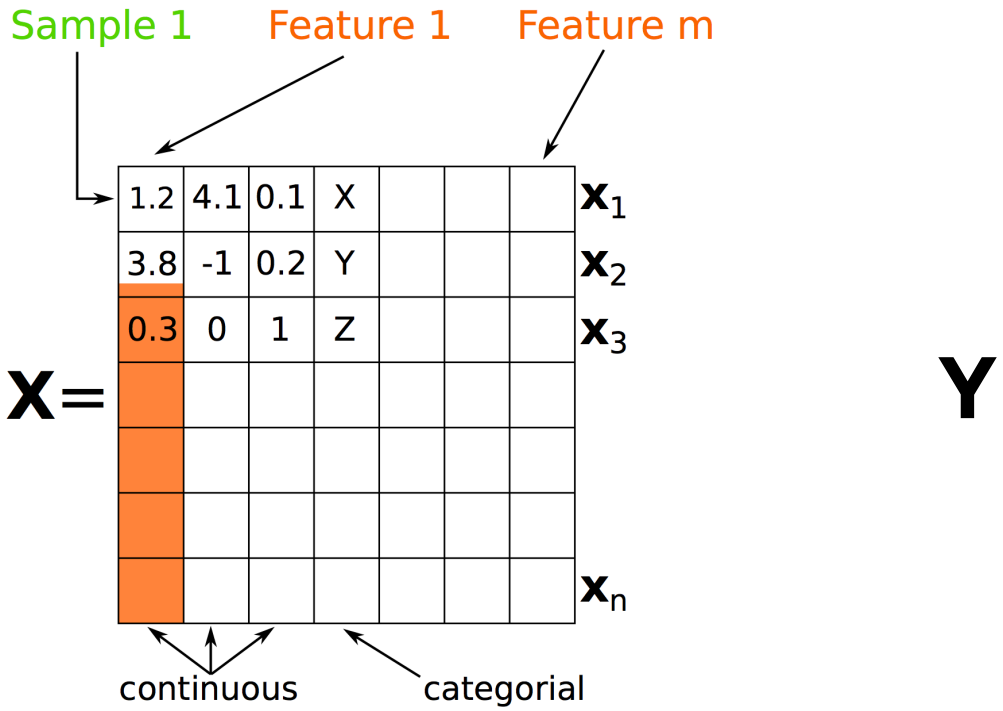
Technische Universität München

+

Johns Hopkins
Whiting School of Engineering

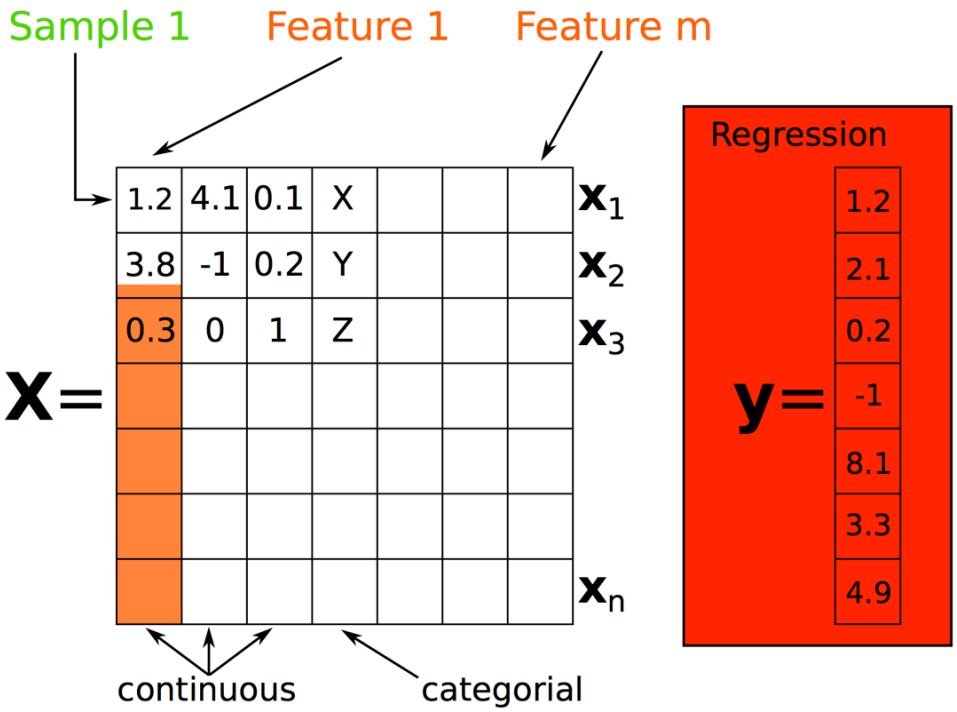
Definition and Problem Statement

- A training set $\{x_i, y_i\}$ is comprised of n samples.
- Every training sample x_i consists of m features $\left(x_{i1}, x_{i2}, ..., x_{im}\right)^T$ and is associated with output y_i.
- Let X indicate a matrix, where every row is a sample and every column is a variable/feature. Features can be either continuous and discrete.



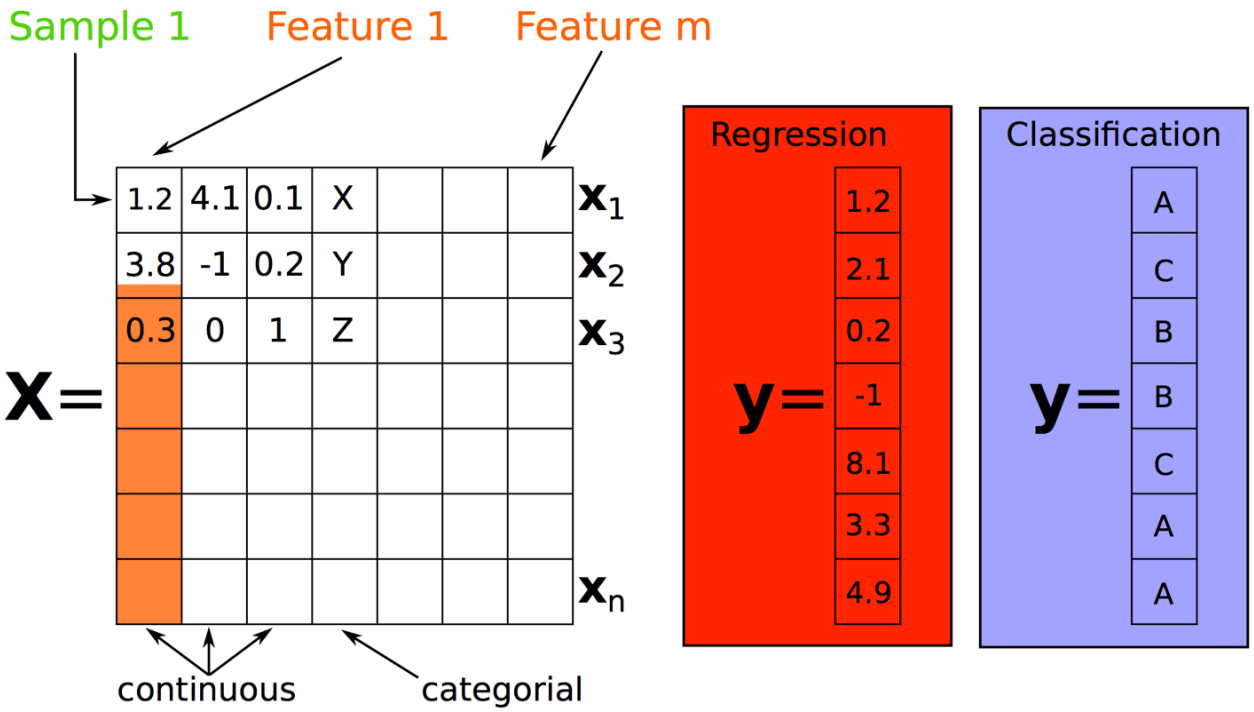
Definition and Problem Statement

- Besides features, output can also be either continuous or discrete. It is a regression problem if the output is continuous



Definition and Problem Statement

- Besides features, output can be either continuous or discrete. It is a regression problem if the output is continuous and a classification problem if the output is discrete.



Definition and Problem Statement

Assumption

There is a function $f(X)$ that relates the features $\left(x_{i1}, x_{i2}, \ldots, x_{im}\right)^T$ to the output y such that $y = f(X)$

Goal

We seek to find a good approximation $\hat{f}(X)$ to the function $f(X)$

If linear model -> linear classifier

Ordinary Linear Regression

Technische Universität München

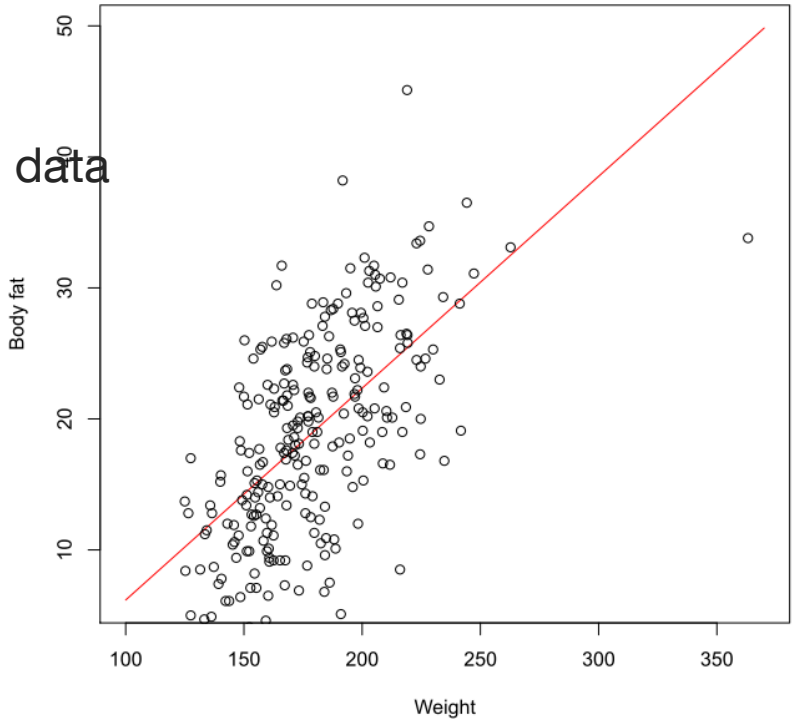
JOHNS HOPKINS
WHITING SCHOOL
of ENGINEERING

Ordinary Linear Regression

- Define the linear model

$$y_i = w_0 + w_1 x_{i1} + \ldots + w_m x_{im} + \varepsilon = w_0 + \mathbf{x}_i^T \mathbf{w} + \varepsilon$$

- The parameters w_i are coefficients or weights of the features and are to be estimated from the training data;
- w_0 is a bias term, also estimated from data
- The error term ε are Gaussian independently identically distributed.



Ordinary Linear Regression – Least Squares Fitting

- Choose the square error loss function

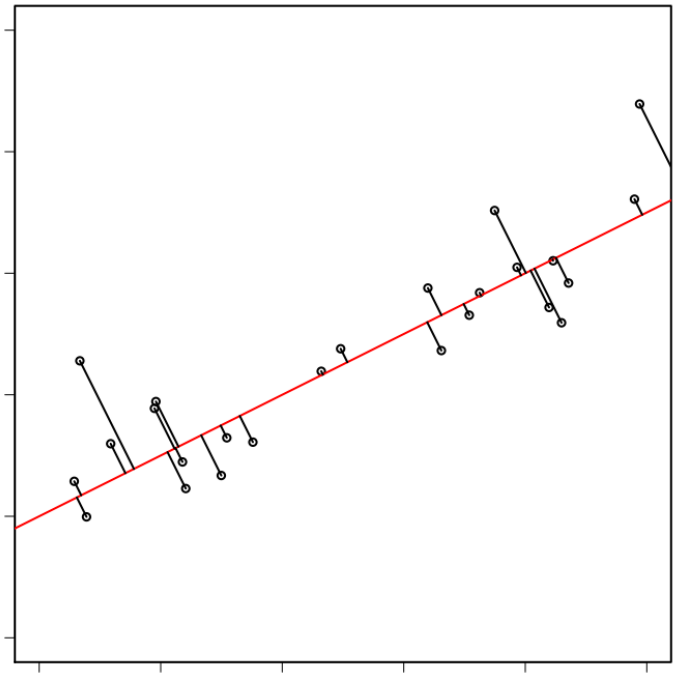
$$L = \left(y_i - \hat{f}\left(x_i \right) \right)^2$$

- Set the partial derivative of loss function to zero

$$\frac{\partial L}{\partial w} = 0$$

$$\hat{w} = \left(X^T X \right)^{-1} X^T y$$

- X must have full column rank
- Features have to be uncorrelated



- Regression is performed using

$$\hat{f}\left(x_1, x_2, \ldots, x_m \right) = \hat{w}_0 + \hat{w}_1 x_1 + \hat{w}_2 x_2 + \ldots + \hat{w}_m x_m$$

Loss Function

- Square error (L2) loss

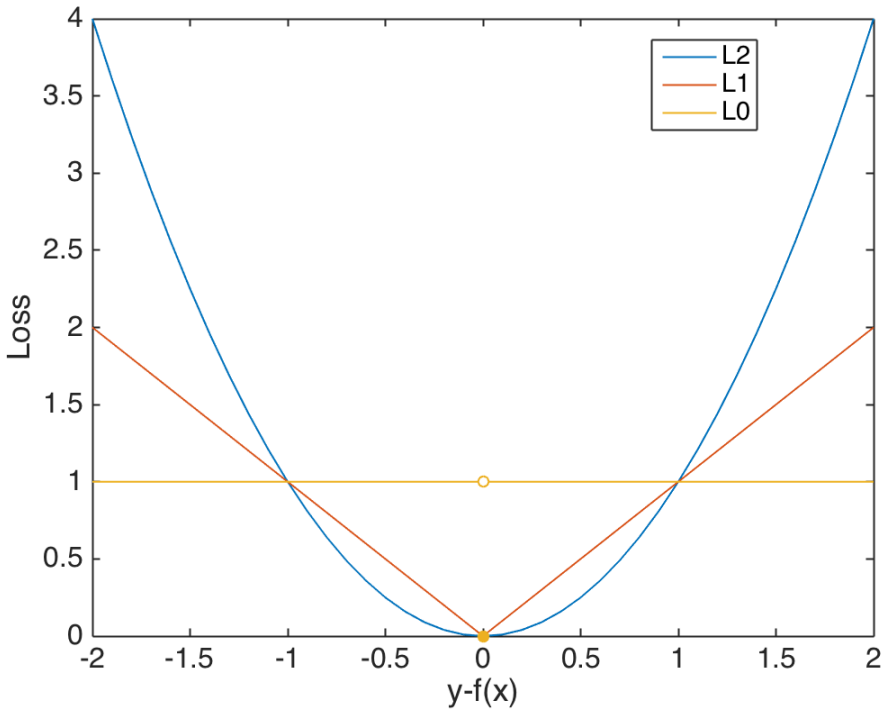
$$\left(y_i - \hat{f}\left(x_i\right)\right)^2$$

- Lasso (L1) loss

$$\left|y_i - \hat{f}\left(x_i\right)\right|$$

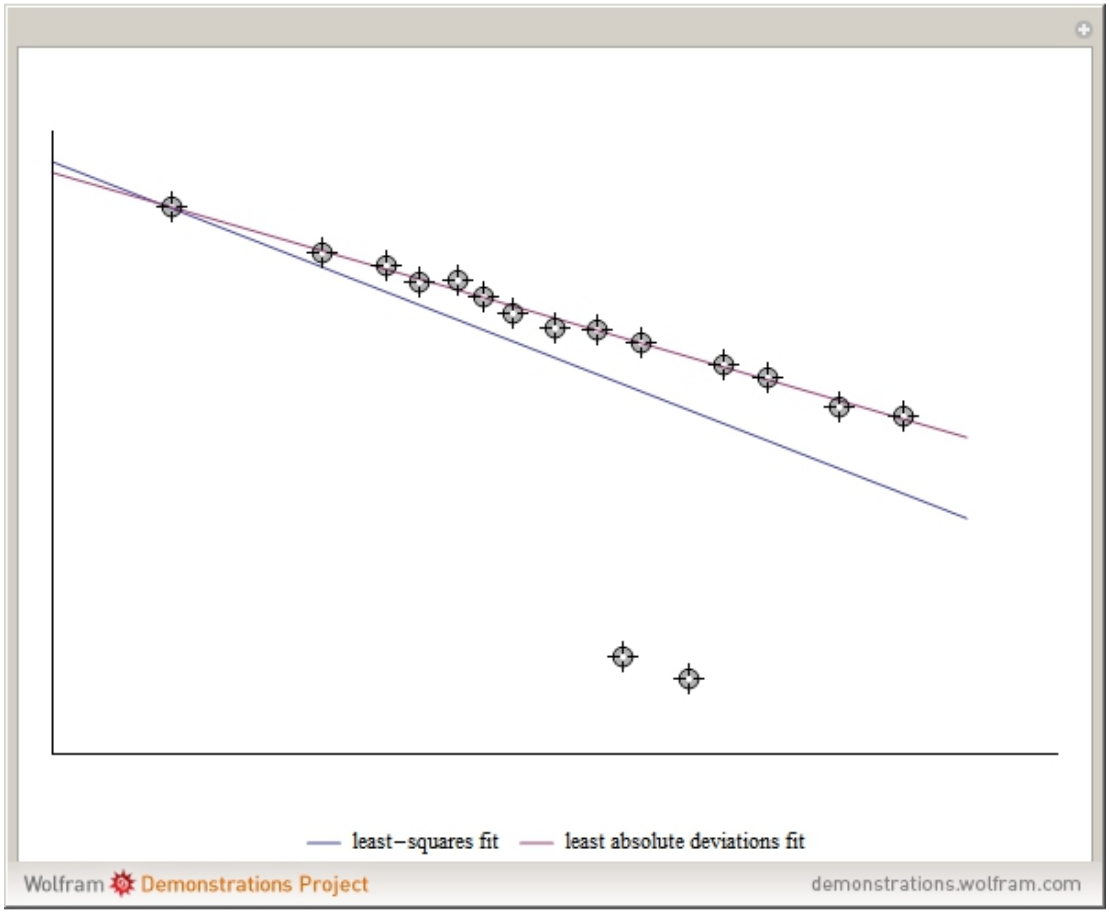
- 0-1 (L0) loss

$$I\left(y_i \neq \hat{f}\left(x_i\right)\right)$$



Loss Function

- Compared to L2 loss, L1/L0 loss is more robust to outliners



Optimal Separating Hyperplanes

Linear Classification

- A training set $\{x_i, y_i\}$ is comprised of n samples.

- Find a hyperplane

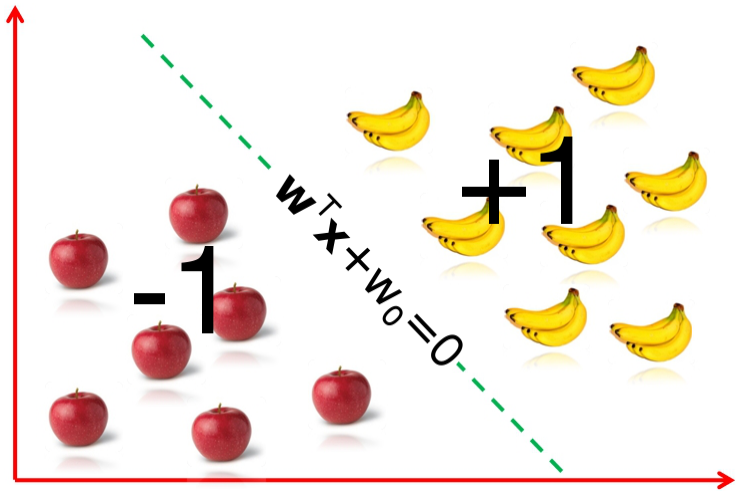
$$h = \mathbf{w}^T \mathbf{x} + w_0 = 0$$

Subject to constraints

$$\mathbf{w}^T \mathbf{x} + w_0 > 0 \text{ if } y_i = +1$$
$$\mathbf{w}^T \mathbf{x} + w_0 < 0 \text{ if } y_i = -1$$

- Decision function

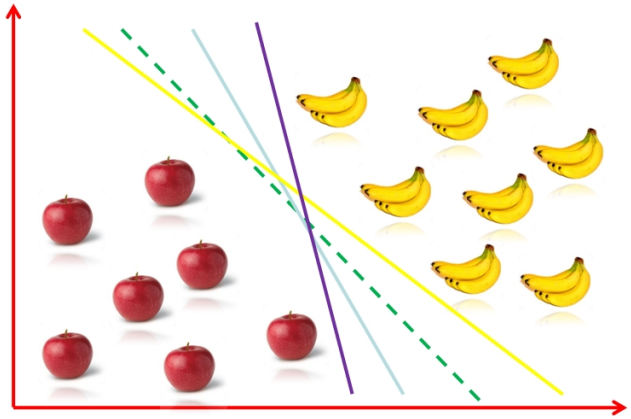
$$f(x) = \text{sign}\left(\mathbf{w}^T \mathbf{x} + w_0\right)$$



Linear Classification

Empirical Risk Minimization

$$[\hat{\mathbf{w}}, \hat{w}_0] = \text{argmin}_{\mathbf{w},w_0} \left(\frac{1}{n} \sum_{i=1}^{n} \left(y_i \neq \text{sign}(\mathbf{w}^T \mathbf{x}_i + w_0) \right) \right)$$



Motivation of SVM:
select ***optimal*** separating hyperplane

Evaluation Measure

Classification Evaluation

- True positive (TP) = positive samples correctly classified as belonging to the positive class
- False positive (FP) = negative samples misclassified as belonging to the positive class
- True negative (TN) = negative samples correctly classified as belonging to the negative class
- False negative (FN) = positive sample misclassified as belonging to the negative class

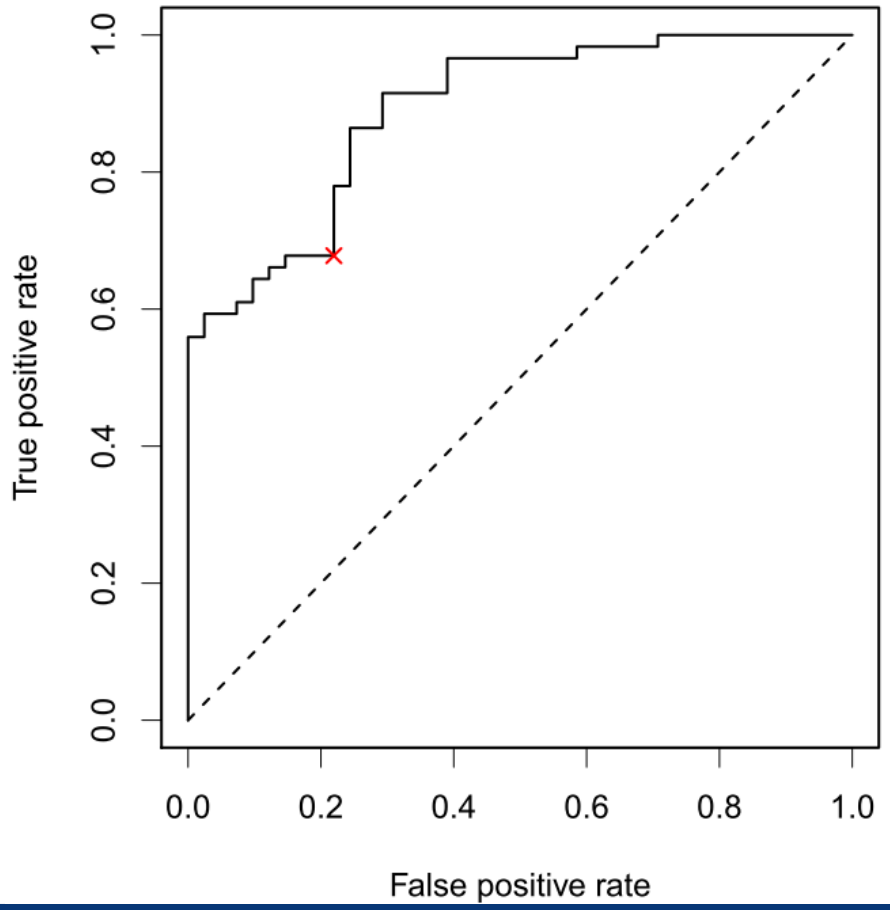
		Ground Truth	
		Class A	**Class B**
Prediction	**Class A**	True positive	False positive Type I Error (α)
	Class B	False negative Type II Error (β)	True negative

Classification Evaluation

- Accuracy = $\frac{TP + TN}{TP + FP + TN + FN}$
- Error rate = 1 – Accuracy
- Sensitivity (true positive rate or recall) = $\frac{TP}{TP + FN}$
- Specificity (true negative rate) = $\frac{TN}{TN + FP}$
- False negative rate = 1 – Sensitivity
- False positive rate = 1 - Specificity

Receiver Operating Characteristics (ROC) Curve

- Binary classifier returns probability or score that represents the degree to which class an instance belongs to



Logistic Regression

Technische Universität München

JOHNS HOPKINS
WHITING SCHOOL
of ENGINEERING

Logistic Regression - Definition

- Consider a binary classification problem where $y_i \in \{0,1\}$
- If $y_i = 1$, the *i*-th sample belongs to the **positive class**, otherwise to the **negative class**.
- Model the probability of sample x_i belong to the positive class

$$\pi_i = P(y_i = 1 \mid x_{i1}, \ldots, x_{im})$$

- Remember the linear model defined as

$$\eta_i = w_0 + w_1 x_{i1} + \ldots + w_m x_{im}$$

- **How to connect the probability π_i to the linear predictor η_i ?**

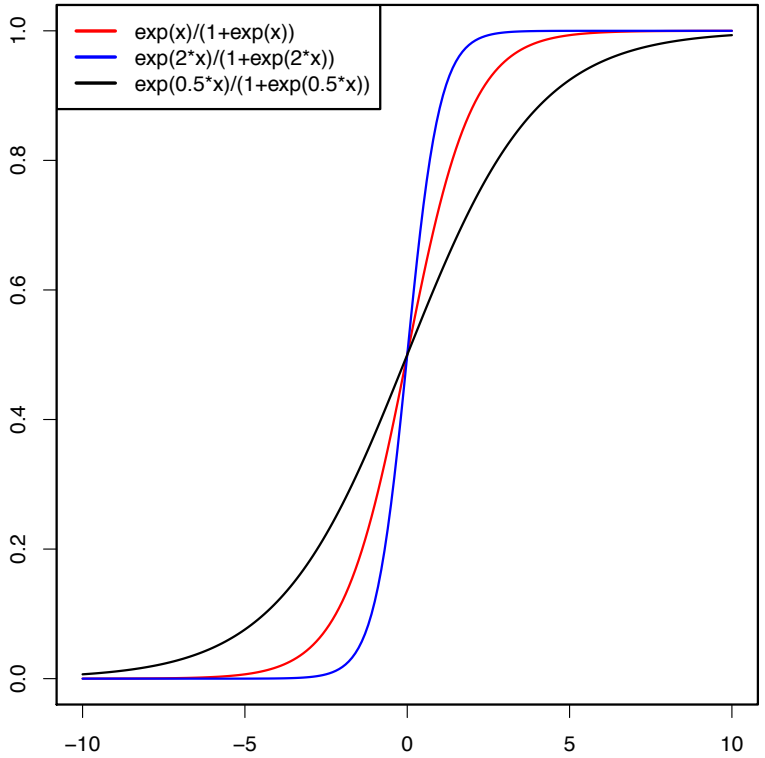
Logistic Regression – Response and link function

- Probability π_i is connected to the linear predictor by the **logistic function**

$$\pi_i = h(\eta_i) = \frac{\exp(\eta_i)}{1+\exp(\eta_i)}$$

- The logistic function is called **response function** and its inverse – the logit function – **link function**

$$h^{-1}(x) = \log\left(\frac{x}{1-x}\right)$$



Logistic Regression – Log-Odds Ratio

- The model is linear with respect to the log-odds:

$$\pi_i = \frac{\exp(\eta_i)}{1+\exp(\eta_i)} \Leftrightarrow \log\left(\frac{\pi_i}{1-\pi_i}\right) = \log\frac{P(y_i=1\,|\,x_i)}{P(y_i=0\,|\,x_i)} = \eta_i$$

- Coefficients indicate by how much the odds change when the value of the corresponding feature is increased by 1

$$\frac{P(y_i=1\,|\,x_{i1},\ldots)}{P(y_i=0\,|\,x_{i1},\ldots)} \Big/ \frac{P(y_i=1\,|\,x_{i1}+1,\ldots)}{P(y_i=0\,|\,x_{i1}+1,\ldots)} = \exp(w_1)$$

Logistic Regression – Log-Odds Ratio

- The coefficient w_j represents the **Log-Odds** of the *j*th feature
- $w_j > 0$, Odds increases, positively correlated with the output
- $w_j < 0$, Odds decreases, negatively correlated with the output
- $w_j = 0$, Odds remain unchanged, uncorrelated with the output
- If $\|w_1\| >> \|w_2\|$ -> feature 1 is more important than feature 2 ??

Logistic Regression - example

Birth weight data contains data from 189 births to determine which of these factors were risk factors for low birth weight (< 2.5 kg) [**?**].

Feature	w / log-odds ratio	Chance
(Intercept)	0.924910	
Age	-0.042784	decreased
Mother's weight (pounds)	-0.015436	decreased
Race = White	0	
Race = Black	1.168452	increased
Race = Other	0.814620	increased
Previous premature labour	1.333970	increased
History of hypertension	1.740511	increased
Smoking during pregnancy	0.858332	increased

Logistic Regression – Maximum Likelihood Estimation

- The likelihood function

$$LF\left(w\right)=\prod_{i=1}^{n}P\left(y_i\middle|x_i\right)=\prod_{i=1}^{n}\pi_i^{y_i}\left(1-\pi_i\right)^{1-y_i}$$

- The log-likelihood function

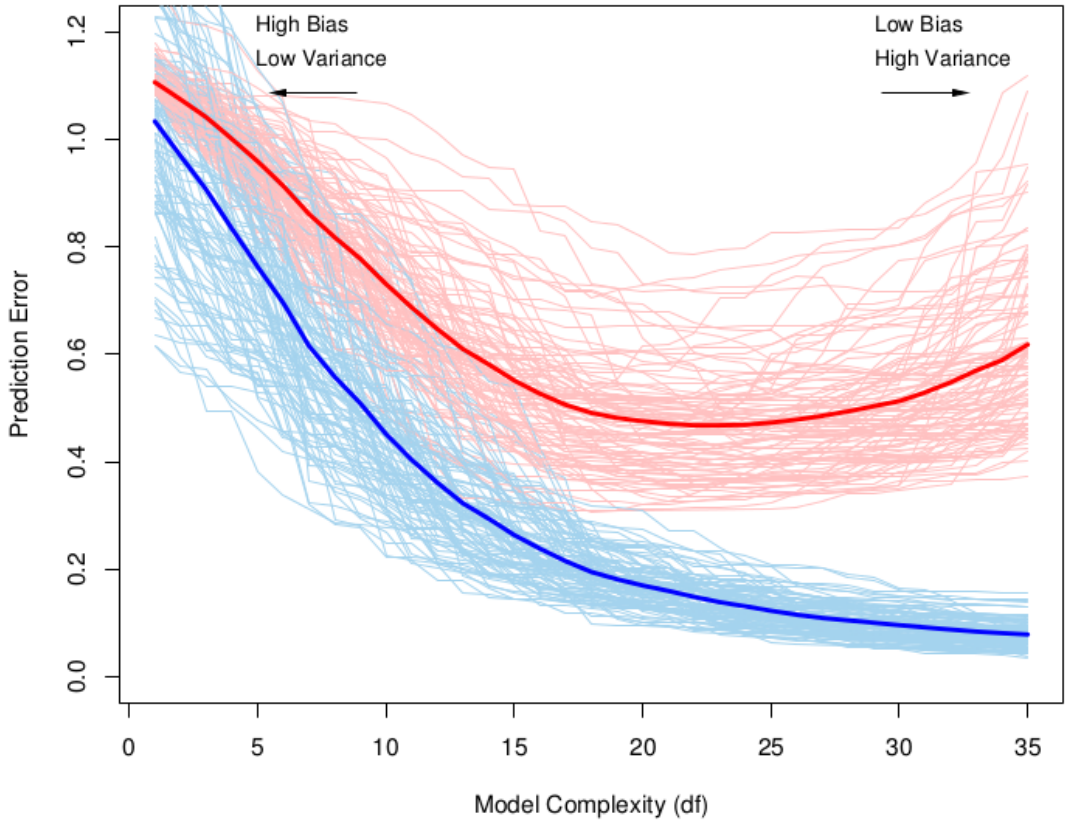
$$\log LF(w)=\sum_{i=1}^{n}y_i\log\left(\pi_i\right)+\left(1-y_i\right)\log\left(1-\pi_i\right)$$

- Maximum likelihood estimation (MLE)

$$\hat{w}=\arg\max_{w}\log LF\left(w\right)$$

Logistic Regression – Regularisation

- Bias-variance trade-off



Training error consistently decreases with increasing model complexity, whereas Testing error starts to increase again.

Logistic Regression – Regularisation

- The logistic loss

$$L\left(w^T x, y\right) = \log(1 + \exp(-w^T x \cdot y))$$

- The regularised logistic regression

$$\hat{w} = \arg\min_{w} L\left(w^T x, y\right) + \lambda h(w)$$

- L2-norm regularisation

$$h(w) = \left\|w\right\|_2 = w^T w$$

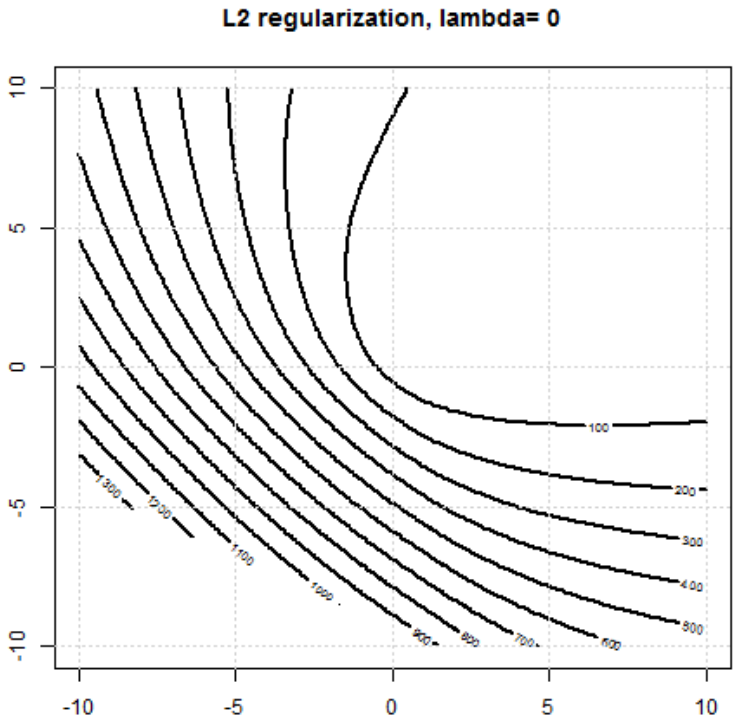
- L1-norm regularisation

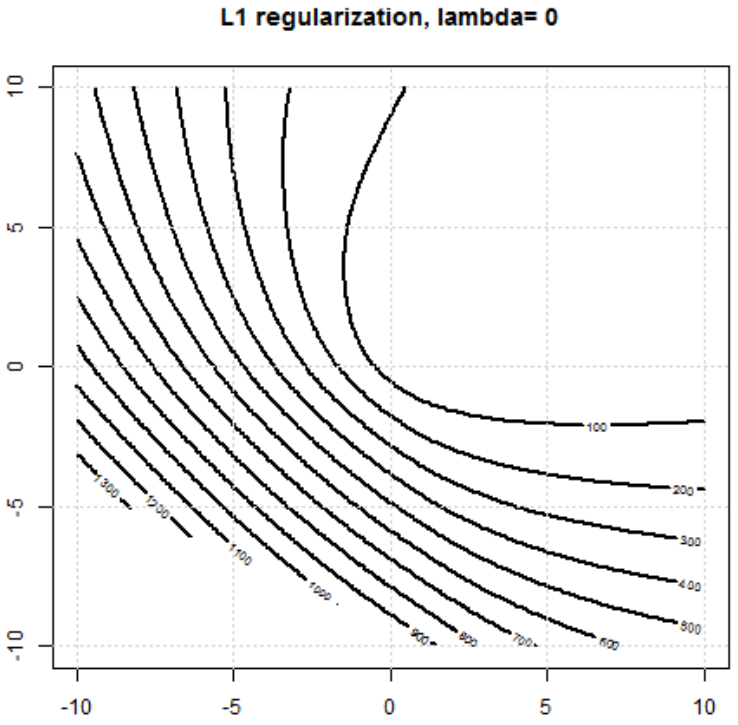
$$h(w) = \left\|w\right\|_1$$

Logistic Regression – Regularisation

- The regularised logistic regression

$$\hat{w} = \arg\min_{w} L\left(w^T x, y\right) + \lambda h(w)$$





Logistic Regression – Model Selection

- Likelihood-ratio Test

$$D = -2\log\left(\frac{\text{likelihood reduced model}}{\text{likelihood full model}}\right).$$

- D can be approximated by a Chi-squared distribution with degrees of freedom (df) equal to the difference in the number of features considered.

- Hence, the p-value can be calculated using the upper incomplete gamma function:

$$p = \Gamma(a,z) = \frac{1}{\Gamma(a)}\int_z^\infty t^{a-1}e^{-t}\,dt$$

$$a = df\,/\,2, t = D\,/\,2$$

- If p<0.05, we can reject he null-hypothesis that the reduced model performs as well as the full model

Logistic Regression – Exercise

- South African Heart Disease

The data set `SAheart` is a subset of the Coronary Risk-Factor Study (CORIS) baseline survey, carried out in three rural areas of the Western Cape, South Africa. The aim of the study was to establish the intensity of ischemic heart disease risk factors in that high-incidence region. The data represents white males between 15 and 64, and the response variable is the presence or absence of myocardial infarction (`chd`) at the time of the survey. The data consists of 160 cases, 302 controls and 9 features. The features are systolic blood pressure (`sbp`), cumulative tobacco in kg (`tobacco`), low density lipoprotein cholesterol (`ldl`), adiposity (`adiposity`), family history of heart disease (`famhist`), type-A behaviour (`typea`), obesity (`obesity`), current alcohol consumption (`alcohol`), age at onset (`age`)

Logistic Regression – Exercise

- South African Heart Disease

  a) Create a model that contains only the intercept (**null model**), i.e. no features are considered.

```
mat_contents = sio.loadmat('SAheart.mat')
SAheart = mat_contents['SAheart']
X = SAheart[:,0:9]
Y = SAheart[:,9]
feature_name = ['sbp','tobacco','Idl','adiposity','famhist','typea','obesity','alcohol','age']

logreg_null = linear_model.LogisticRegression(C=1e6,solver='newton-cg',intercept_scaling=1e3)

# we fit the null model.
logreg_null.fit(np.zeros(Y.size).reshape(-1,1), Y)
```

Logistic Regression – Exercise

- South African Heart Disease

b) Create multiple models each considering a single feature. Note that `famhist` is a categorical feature which has to be converted to numbers first.

```
for i in range(X.shape[1]):
    logreg_sf.fit(X[:,i].reshape(-1,1),Y)
```

Logistic Regression – Feature Selection

- South African Heart Disease

c) Create a function `likelihood_ratio_test` implementing the likelihood-ratio test which takes the log-likelihood of the full model and the reduced model (Section 1.1). Use this function to compare the *single feature models* to the *null model*. Which feature yields the most significant improvement over the null model? Make sure you consider the p-value of the likelihood-ratio test.

```
def likelihood_ratio_test(LLF_full,LLF_reduced,df):
    D = -2*(LLF_reduced-LLF_full)
    p = sspecial.gammaincc(df/2,D/2)
    return p
```

Logistic Regression – Feature Selection

- South African Heart Disease

d) Create a model which considers multiple features by starting with the null model and adding one additional feature at a time. To determine which feature to add, use the p-value as returned by the likelihood-ratio test. Extended models with one additional feature, where the p-value is greater than 0.05, should not be considered. In each step choose the model with the smallest p-value. Continue until all features have been selected or the model cannot be improved significantly any more. Print all selected features.

```
Likelyhood selected features:
intercept: -4.11859447037
age : 0.0464558440418
tobacco : 0.080848897373
famhist : 0.925862883712
Idl : 0.177595334993
adiposity : -0.00929820367955
```

Logistic Regression – Feature Selection

- South African Heart Disease

e) L1 (lasso) regularization can also be used for feature selection. Consider a full model with all features as input, penalized with the L1 norm of coefficients (try regularization parameter `C` in the range of $0.01 - 0.1$). Features with an non-zero coefficient are important for the classification. Compare the Lasso-selected features to the features selected by p-values. Please note Lasso-feature selection requires a standardization of features that each feature has a zero mean and a unit standard derivation (e.g.using Sklearn built-in function `sklearn.preprocessing.scale`)

```
L1 selected features:

intercept: -0.728330817135

age : 0.479362009804

famhist : 0.256833232171

tobacco : 0.209469607939

Idl : 0.176112199506

typea : 0.0706683038816
```

Logistic Regression – Feature Selection

- South African Heart Disease

```
Likelyhood selected features:
intercept: -4.11859447037
age : 0.0464558440418
tobacco : 0.080848897373
famhist : 0.925862883712
Idl : 0.177595334993
adiposity : -0.00929820367955
```

```
L1 selected features:
intercept: -0.728330817135
age : 0.479362009804
famhist : 0.256833232171
tobacco : 0.209469607939
Idl : 0.176112199506
typea : 0.0706683038816
```