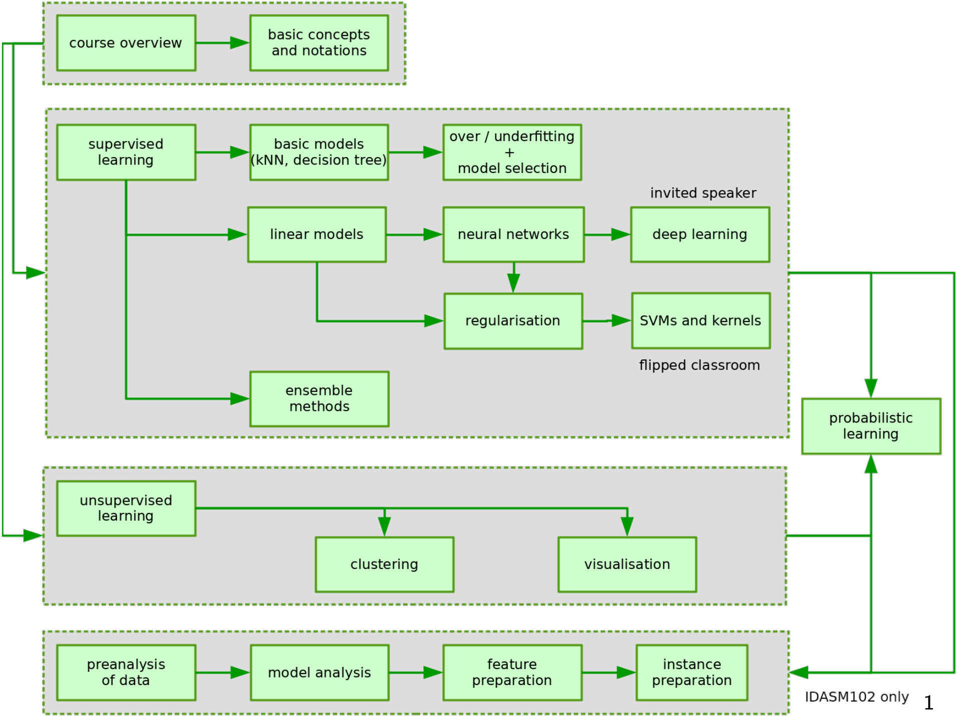
Machine Learning: Lesson 6

Linear Models for Regression and Classification

Benoît Frénay - Faculty of Computer Science

UNIVERSITÉ DE NAMUR


course overview
basic concepts and notations
supervised learning
basic models (kNN, decision tree)
over / underfitting + model selection
invited speaker
linear models
neural networks
deep learning
regularisation
SVMs and kernels
flipped classroom
ensemble methods
probabilistic learning
unsupervised learning
clustering
visualisation
preanalysis of data
model analysis
feature preparation
instance preparation
IDASM102 only

Outline of this Lesson

- regression with linear models
- classification with linear models
- outliers in regression and classification

Regression with Linear Models

Linear Models for Regression

Available data

a set of n training data $\mathcal{D} = \{(\mathbf{x}_i, t_i)\}$ where

- $\mathbf{x} \in \Re^d$ is a vector of d continuous features
- $\mathbf{t} \in \Re$ is a continuous target value

Linear modelling

assumption = feature values in $\mathbf{x}$ and the target value t are linearly related

$$f(x_1, \ldots, x_n) = w_1 x_1 + \cdots + w_d x_d + w_0$$

each weight w_j models the contribution of feature x_j to the target value t

Linear Models for Regression

Available data

a set of n training data $\mathcal{D} = \{(\mathbf{x}_i, t_i)\}$ where

- $\mathbf{x} \in \Re^d$ is a vector of d continuous features
- $\mathbf{t} \in \Re$ is a continuous target value

Linear modelling

assumption = feature values in $\mathbf{x}$ and the target value t are linearly related

$$f(x_1, \ldots, x_n) = w_1 x_1 + \cdots + w_d x_d + w_0$$

each weight w_j models the contribution of feature x_j to the target value t

Boston Housing Prices Dataset ($n = 506$ and $d = 13$)

crim	per capita crime rate by town
zn	proportion of residential land zoned for lots over 25,000 sq. ft.
indus	proportion of non-retain business acres per town
chas	Charles River dummy variable (= 1 if tract bounds river; 0 otherwise)
nox	nitrogen oxides concentration (parts per million)
rm	average number of rooms per dwelling
age	proportion of owner-occupied units built prior to 1940
dis	weighted mean of distances to five Boston employment centers
rad	index of accessibility to radial highways
tax	full-value property-tax rate per $10,000
ptratio	pupil-teacher ratio by town
black	1000(Bk - 0.63)^2, where Bk is the proportion of blacks by town
lstat	lower status of the population (percent)
medv	median value of owner-occupied homes in $1000s

Result of linear regression (top 5 features and mean error $\approx$ \$3200)

$$f\left(x_{\text{crim}} \ldots | w_{\text{crim}} \ldots\right) = 2.7\, x_{\text{chas}} - 17.8\, x_{\text{nox}} + 3.8\, x_{\text{rm}} - 1.5\, x_{\text{dis}} - 0.9\, x_{\text{ptratio}} + 36.49$$

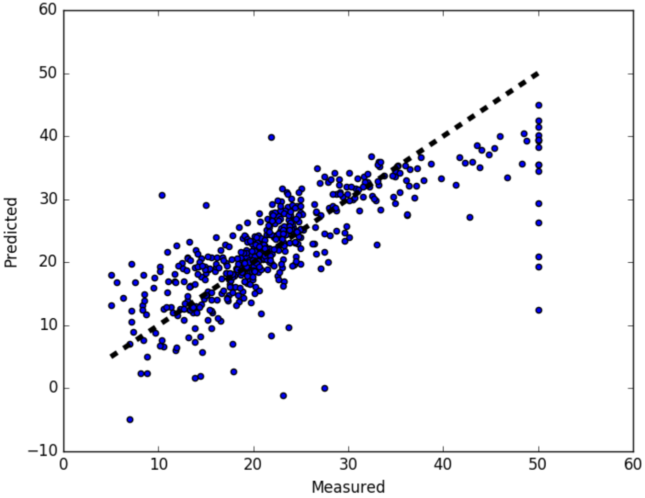
Boston Housing Prices Dataset ($n = 506$ and $d = 13$)

crim	per capita crime rate by town
zn	proportion of residential land zoned for lots over 25,000 sq. ft.
indus	proportion of non-retain business acres per town
chas	Charles River dummy variable (= 1 if tract bounds river; 0 otherwise)
nox	nitrogen oxides concentration (parts per million)
rm	average number of rooms per dwelling
age	proportion of owner-occupied units built prior to 1940
dis	weighted mean of distances to five Boston employment centers
rad	index of accessibility to radial highways
tax	full-value property-tax rate per $10,000
ptratio	pupil-teacher ratio by town
black	$1000(\text{Bk} - 0.63)^2$, where Bk is the proportion of blacks by town
lstat	lower status of the population (percent)
medv	median value of owner-occupied homes in $1000s

Result of linear regression (top 5 features and mean error ≈ $3200)

$$f\left(x_{\text{crim}} \ldots | w_{\text{crim}} \ldots\right) = 2.7\, x_{\text{rm}} - 3.1\, x_{\text{dis}} + 2.7\, x_{\text{rad}} - 2.1\, x_{\text{ptratio}} - 3.7\, x_{\text{lstat}} + 22.53$$

Boston Housing Prices Dataset ($n = 506$ and $d = 13$)



source: http://scikit-learn.org/stable/auto_examples/plot_cv_predict.html

Optimising Linear Models for Regression

Criterion: mean square error

in practice, it is often impossible to exactly reproduce the target values

- the relationship between $\mathbf{x}$ and t may be partially non-linear
- t is often affected by some noise (measurement, transcription, etc.)

one solution is to minimise the mean square error

$$\mathrm{MSE} = \frac{1}{n}\sum_{i=1}^{n}\left(t_i - f(x_{i1}, \ldots, x_{id})\right)^2$$

Algorithms for MSE optimisation

linear models can be optimised for regression w.r.t. the MSE

- pseudo-inverse method: analytical, exact solution in one step
- iterative algorithms like (stochastic) gradient descent

Optimising Linear Models for Regression

Criterion: mean square error

in practice, it is often impossible to exactly reproduce the target values

- the relationship between $\mathbf{x}$ and t may be partially non-linear
- t is often affected by some noise (measurement, transcription, etc.)

one solution is to minimise the mean square error

$$\text{MSE} = \frac{1}{n}\sum_{i=1}^{n}\left(t_i - f(x_{i1}, \ldots, x_{id})\right)^2$$

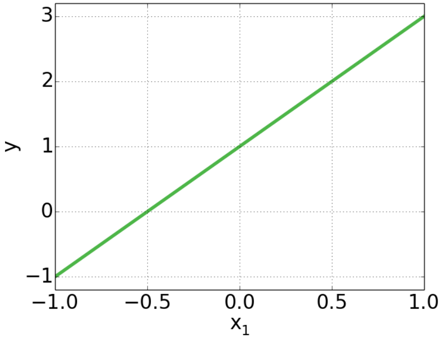
Algorithms for MSE optimisation

linear models can be optimised for regression w.r.t. the MSE

- pseudo-inverse method: analytical, exact solution in one step
- iterative algorithms like (stochastic) gradient descent

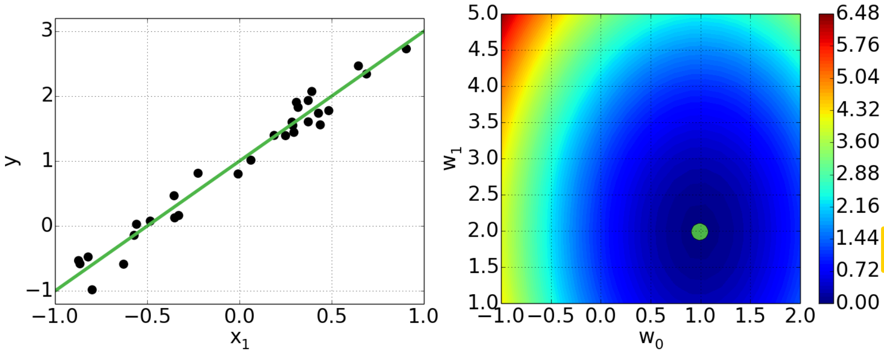
Mean Square Loss: Illustration on a Linear Problem

$f(x) = 2x + 1 + \epsilon$ with $\epsilon \sim \mathcal{N}(0, 0.2)$



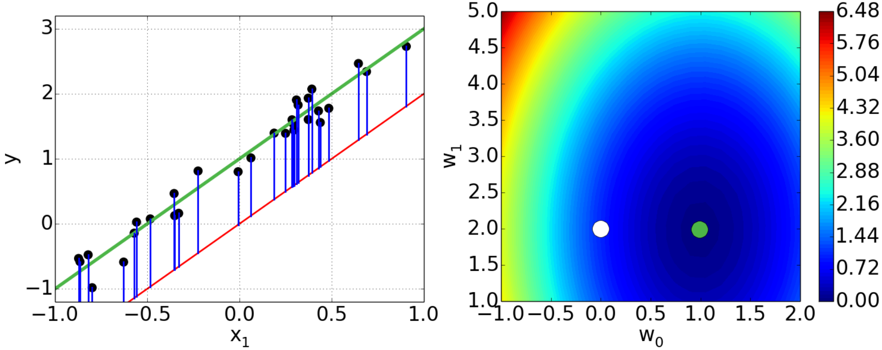
Mean Square Loss: Illustration on a Linear Problem

$f(x) = 2x + 1 + \epsilon$ with $\epsilon \sim \mathcal{N}(0, 0.2)$ and $n = 30$



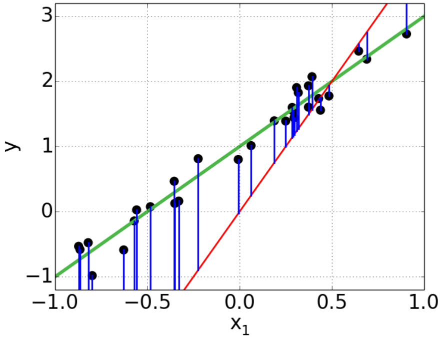
Mean Square Loss: Illustration on a Linear Problem

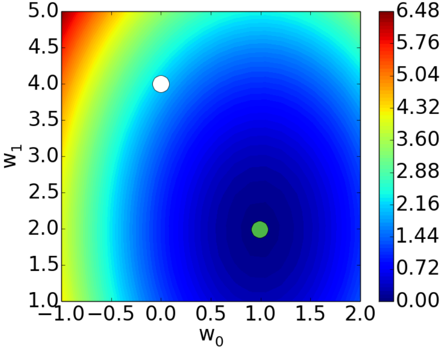
$f(x) = 2x + 1 + \epsilon$ with $\epsilon \sim \mathcal{N}(0, 0.2)$ and $n = 30 \Rightarrow$ MSE $= 1.021$



Mean Square Loss: Illustration on a Linear Problem

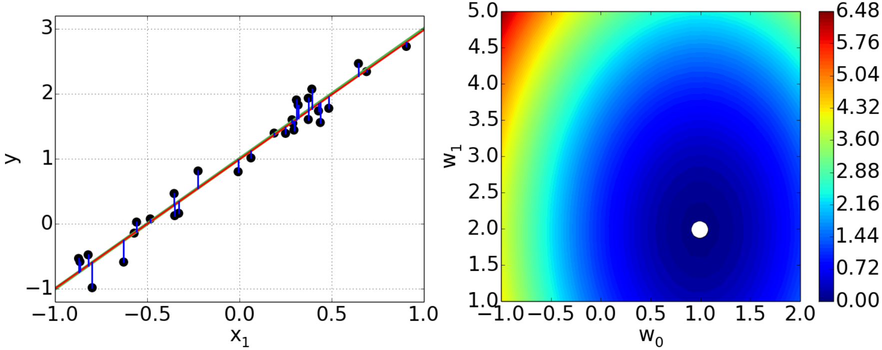
$f(x) = 2x + 1 + \epsilon$ with $\epsilon \sim \mathcal{N}(0, 0.2)$ and $n = 30 \Rightarrow$ MSE $= 2.090$





Mean Square Loss: Illustration on a Linear Problem

$f(x) = 2x + 1 + \epsilon$ with $\epsilon \sim \mathcal{N}(0, 0.2)$ and $n = 30 \Rightarrow$ MSE $= 0.035$



Linear Regression: Optimising Linear Models in one Step

Linear regression / ordinary least squares / pseudo-inverse method

Input: dataset $\mathcal{D} = \{(\mathbf{x}_i, t_i)\}$ in matrix/vectorial form as $\mathbf{X}$ and $\mathbf{t}$

Output: optimal weights for linear regression (w.r.t. MSE)

return $\mathbf{w} = \arg\min_{\mathbf{w}} \frac{1}{n} \sum_{i=1}^{n} (t_i - f(x_{i1}, \ldots, x_{id}))^2 = (\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\mathbf{t}$

Advantages

- give the exact solution in one step, standard in statistics
- fast and cheap for low dimension data, easy to implement

Limitations

- does not scale with dimension (time $= \mathcal{O}(d^3)$, memory $= \mathcal{O}(d^2)$)
- the matrix $\mathbf{X}^T\mathbf{X}$ may be ill-conditioned and impossible to invert

Linear Regression: Optimising Linear Models in one Step

Linear regression / ordinary least squares / pseudo-inverse method

Input: dataset $\mathcal{D} = \{(\mathbf{x}_i, t_i)\}$ in matrix/vectorial form as $\mathbf{X}$ and $\mathbf{t}$
Output: optimal weights for linear regression (w.r.t. MSE)

return $\mathbf{w} = \arg\min_{\mathbf{w}} \frac{1}{n} \sum_{i=1}^{n} \left(t_i - f(x_{i1}, \ldots, x_{id})\right)^2 = \left(\mathbf{X}^T\mathbf{X}\right)^{-1}\mathbf{X}^T\mathbf{t}$

Advantages

- give the exact solution in one step, standard in statistics
- fast and cheap for low dimension data, easy to implement

Limitations

- does not scale with dimension (time $= \mathcal{O}(d^3)$, memory $= \mathcal{O}(d^2)$)
- the matrix $\mathbf{X}^T\mathbf{X}$ may be ill-conditioned and impossible to invert

Linear Regression: Optimising Linear Models in one Step

Linear regression / ordinary least squares / pseudo-inverse method

Input: dataset $\mathcal{D} = \{(\mathbf{x}_i, t_i)\}$ in matrix/vectorial form as $\mathbf{X}$ and $\mathbf{t}$
Output: optimal weights for linear regression (w.r.t. MSE)

return $\mathbf{w} = \arg\min_{\mathbf{w}} \frac{1}{n} \sum_{i=1}^{n} \left(t_i - f(x_{i1}, \ldots, x_{id})\right)^2 = \left(\mathbf{X}^T \mathbf{X}\right)^{-1} \mathbf{X}^T \mathbf{t}$

Advantages

- give the exact solution in one step, standard in statistics
- fast and cheap for low dimension data, easy to implement

Limitations

- does not scale with dimension (time $= \mathcal{O}(d^3)$, memory $= \mathcal{O}(d^2)$)
- the matrix $\mathbf{X}^T \mathbf{X}$ may be ill-conditioned and impossible to invert

Linear Regression: Optimising Linear Models in one Step

Linear regression / ordinary least squares / pseudo-inverse method

Input: dataset $\mathcal{D} = \{(\mathbf{x}_i, t_i)\}$ in matrix/vectorial form as $\mathbf{X}$ and $\mathbf{t}$
Output: optimal weights for linear regression (w.r.t. MSE)

return $\mathbf{w} = \arg\min_{\mathbf{w}} \frac{1}{n} \sum_{i=1}^{n} (t_i - f(x_{i1}, \ldots, x_{id}))^2 = (\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\mathbf{t}$

Advantages

- give the exact solution in one step, standard in statistics
- fast and cheap for low dimension data, easy to implement

Limitations

- does not scale with dimension (time $= \mathcal{O}(d^3)$, memory $= \mathcal{O}(d^2)$)
- the matrix $\mathbf{X}^T\mathbf{X}$ may be ill-conditioned and impossible to invert

Linear Regression: Optimising Linear Models in one Step

Linear regression / ordinary least squares / pseudo-inverse method

Input: dataset $\mathcal{D} = \{(\mathbf{x}_i, t_i)\}$ in matrix/vectorial form as $\mathbf{X}$ and $\mathbf{t}$
Output: optimal weights for linear regression (w.r.t. MSE)

return $\mathbf{w} = \arg\min_{\mathbf{w}} \frac{1}{n}\sum_{i=1}^{n}\left(t_i - f(x_{i1}, \ldots, x_{id})\right)^2 = \left(\mathbf{X}^T\mathbf{X}\right)^{-1}\mathbf{X}^T\mathbf{t}$

Advantages

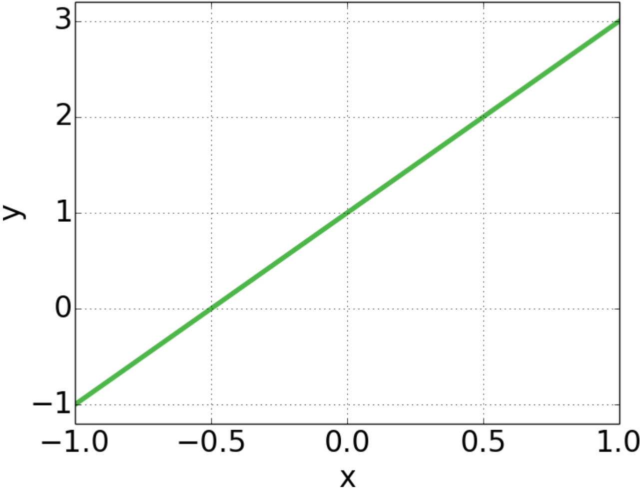
- give the exact solution in one step, standard in statistics
- fast and cheap for low dimension data, easy to implement

Limitations

- does not scale with dimension (time $= \mathcal{O}(d^3)$, memory $= \mathcal{O}(d^2)$)
- the matrix $\mathbf{X}^T\mathbf{X}$ may be ill-conditioned and impossible to invert

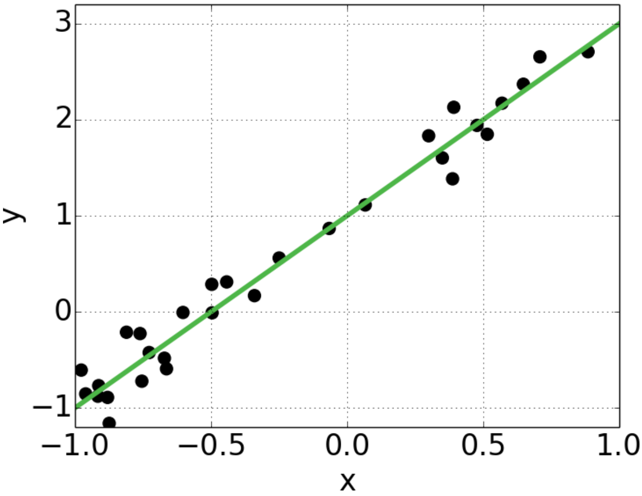
Example of Linear Regression

$f(x) = 2x + 1 + \epsilon$ with $\epsilon \sim \mathcal{N}(0, 0.2)$



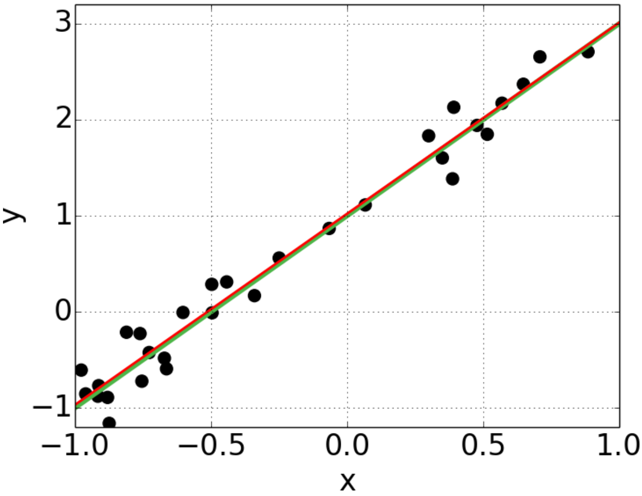
Example of Linear Regression

$f(x) = 2x + 1 + \epsilon$ with $\epsilon \sim \mathcal{N}(0, 0.2)$ and $n = 30$



Example of Linear Regression

$f(x) = 2x + 1 + \epsilon$ with $\epsilon \sim \mathcal{N}(0, 0.2)$ and $n = 30 \Rightarrow \hat{\mathbf{w}} = (1.02, 1.99)$



Application: Diabetes Progression

Task description

- goal: predict the **diabetes progression** one year after baseline
- 442 **diabetes patients** were measured on **10 baseline variables**

Available patient characteristics (features)

1 age

2 sex

3 body mass index (BMI)

4 blood pressure (BP)

5 serum measurement #1

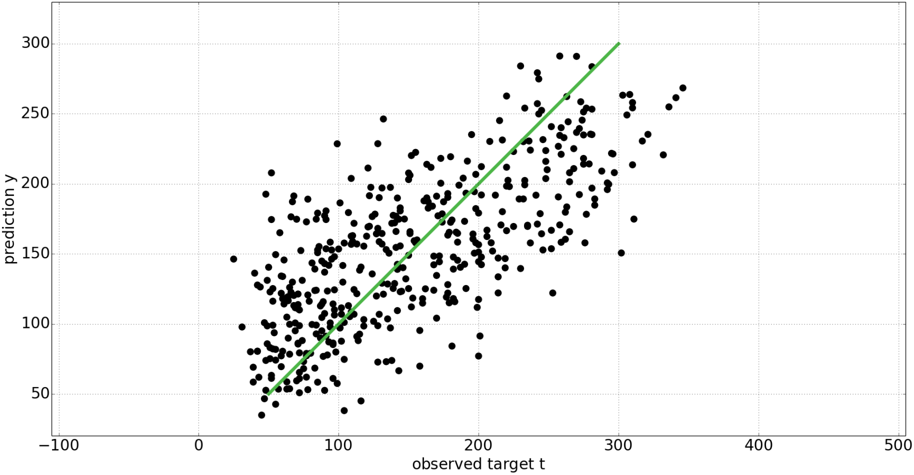
... ...

10 serum measurement #6

Efron, B., Hastie, T., Johnstone, I., Tishirani, R. Least Angle Regression. Annals of Statistics 32 p. 407–499, 2004.

Application: Diabetes Progression

$n = 442$ patients with $d = 10$ features $\Rightarrow$ RMSE $= \sqrt{\text{MSE}} = 53.49$



Probabilistic Interpretation of the MSE

target values are assumed to be polluted by a Gaussian noise $\mathcal{N}\left(0, \sigma_t^2\right)$

$$
\begin{aligned}
\log p\left(\mathbf{t} | \mathbf{X}, \mathbf{w}, \sigma_t\right) &= \sum_{i=1}^{n} \log p\left(t_i | \mathbf{x_i}, \mathbf{w}, \sigma_t\right) \\
&= \sum_{i=1}^{n} \log \mathcal{N}\left(t_i - f(x_{i1}, \ldots, x_{id}) | 0, \sigma_t^2\right) \\
&= \sum_{i=1}^{n} \log \frac{1}{\sqrt{2\pi\sigma_t^2}} \exp\left(-\frac{\left(t_i - f(x_{i1}, \ldots, x_{id})\right)^2}{2\sigma_t^2}\right) \\
&= -\frac{1}{2\sigma_t^2} \sum_{i=1}^{n} \left(t_i - f(x_{i1}, \ldots, x_{id})\right)^2 - \frac{n}{2} \log 2\pi\sigma_t^2 .
\end{aligned}
$$

$$
\text{max. } p\left(\mathbf{t} | \mathbf{X}, \mathbf{w}, \sigma_t\right) \Leftrightarrow \text{min. } \frac{1}{\sigma_t^2} \times \underbrace{\frac{1}{n} \sum_{i=1}^{n} \left(t_i - f(x_{i1}, \ldots, x_{id})\right)^2}_{MSE} + \underbrace{\log \sigma_t^2}_{\sigma_t^2 \text{ term}}
$$

Probabilistic Interpretation of the MSE

target values are assumed to be polluted by a Gaussian noise $\mathcal{N}\left(0, \sigma_t^2\right)$

$$\log p\left(\mathbf{t}|\mathbf{X}, \mathbf{w}, \sigma_t\right) = \sum_{i=1}^{n} \log p\left(t_i|\mathbf{x_i}, \mathbf{w}, \sigma_t\right)$$

$$= \sum_{i=1}^{n} \log \mathcal{N}\left(t_i - f(x_{i1}, \ldots, x_{id})|0, \sigma_t^2\right)$$

$$= \sum_{i=1}^{n} \log \frac{1}{\sqrt{2\pi\sigma_t^2}} \exp\left(-\frac{\left(t_i - f(x_{i1}, \ldots, x_{id})\right)^2}{2\sigma_t^2}\right)$$

$$= -\frac{1}{2\sigma_t^2} \sum_{i=1}^{n} \left(t_i - f(x_{i1}, \ldots, x_{id})\right)^2 - \frac{n}{2} \log 2\pi\sigma_t^2.$$

max. $p\left(\mathbf{t}|\mathbf{X}, \mathbf{w}, \sigma_t\right) \Leftrightarrow$ min. $\frac{1}{\sigma_t^2} \times \underbrace{\frac{1}{n} \sum_{i=1}^{n} \left(t_i - f(x_{i1}, \ldots, x_{id})\right)^2}_{MSE} + \underbrace{\log \sigma_t^2}_{\sigma_t^2 \text{ term}}$

Probabilistic Interpretation of the MSE

target values are assumed to be polluted by a Gaussian noise $\mathcal{N}\left(0, \sigma_t^2\right)$

$$
\begin{aligned}
\log p\left(\mathbf{t}|\mathbf{X}, \mathbf{w}, \sigma_t\right) &= \sum_{i=1}^{n} \log p\left(t_i|\mathbf{x_i}, \mathbf{w}, \sigma_t\right) \\
&= \sum_{i=1}^{n} \log \mathcal{N}\left(t_i - f(x_{i1}, \ldots, x_{id})|0, \sigma_t^2\right) \\
&= \sum_{i=1}^{n} \log \frac{1}{\sqrt{2\pi\sigma_t^2}} \exp\left(-\frac{(t_i - f(x_{i1}, \ldots, x_{id}))^2}{2\sigma_t^2}\right) \\
&= -\frac{1}{2\sigma_t^2} \sum_{i=1}^{n} (t_i - f(x_{i1}, \ldots, x_{id}))^2 - \frac{n}{2} \log 2\pi\sigma_t^2.
\end{aligned}
$$

max. $p\left(\mathbf{t}|\mathbf{X}, \mathbf{w}, \sigma_t\right) \Leftrightarrow$ min. $\frac{1}{\sigma_t^2} \times \underbrace{\frac{1}{n} \sum_{i=1}^{n} (t_i - f(x_{i1}, \ldots, x_{id}))^2}_{MSE} + \underbrace{\log \sigma_t^2}_{\sigma_t^2 \text{ term}}$

Probabilistic Interpretation of the MSE

target values are assumed to be polluted by a Gaussian noise $\mathcal{N}\left(0, \sigma_t^2\right)$

$$\begin{aligned}
\log p\left(\mathbf{t}|\mathbf{X}, \mathbf{w}, \sigma_t\right) &= \sum_{i=1}^{n} \log p\left(t_i|\mathbf{x_i}, \mathbf{w}, \sigma_t\right) \\
&= \sum_{i=1}^{n} \log \mathcal{N}\left(t_i - f(x_{i1}, \ldots, x_{id})|0, \sigma_t^2\right) \\
&= \sum_{i=1}^{n} \log \frac{1}{\sqrt{2\pi\sigma_t^2}} \exp\left(-\frac{\left(t_i - f(x_{i1}, \ldots, x_{id})\right)^2}{2\sigma_t^2}\right) \\
&= -\frac{1}{2\sigma_t^2} \sum_{i=1}^{n} \left(t_i - f(x_{i1}, \ldots, x_{id})\right)^2 - \frac{n}{2} \log 2\pi\sigma_t^2.
\end{aligned}$$

$$\text{max. } p\left(\mathbf{t}|\mathbf{X}, \mathbf{w}, \sigma_t\right) \Leftrightarrow \text{min. } \frac{1}{\sigma_t^2} \times \underbrace{\frac{1}{n} \sum_{i=1}^{n} \left(t_i - f(x_{i1}, \ldots, x_{id})\right)^2}_{MSE} + \underbrace{\log \sigma_t^2}_{\sigma_t^2 \text{ term}}$$

Probabilistic Interpretation of the MSE

target values are assumed to be polluted by a Gaussian noise $\mathcal{N}\left(0, \sigma_t^2\right)$

$$
\begin{aligned}
\log p\left(\mathbf{t}|\mathbf{X}, \mathbf{w}, \sigma_t\right) &= \sum_{i=1}^{n} \log p\left(t_i|\mathbf{x_i}, \mathbf{w}, \sigma_t\right) \\
&= \sum_{i=1}^{n} \log \mathcal{N}\left(t_i - f(x_{i1}, \ldots, x_{id})|0, \sigma_t^2\right) \\
&= \sum_{i=1}^{n} \log \frac{1}{\sqrt{2\pi\sigma_t^2}} \exp\left(-\frac{(t_i - f(x_{i1}, \ldots, x_{id}))^2}{2\sigma_t^2}\right) \\
&= -\frac{1}{2\sigma_t^2} \sum_{i=1}^{n} (t_i - f(x_{i1}, \ldots, x_{id}))^2 - \frac{n}{2} \log 2\pi\sigma_t^2.
\end{aligned}
$$

max. $p\left(\mathbf{t}|\mathbf{X}, \mathbf{w}, \sigma_t\right) \Leftrightarrow$ min. $\frac{1}{\sigma_t^2} \times \underbrace{\frac{1}{n} \sum_{i=1}^{n} (t_i - f(x_{i1}, \ldots, x_{id}))^2}_{MSE} + \underbrace{\log \sigma_t^2}_{\sigma_t^2 \text{ term}}$

Probabilistic Interpretation of the MSE

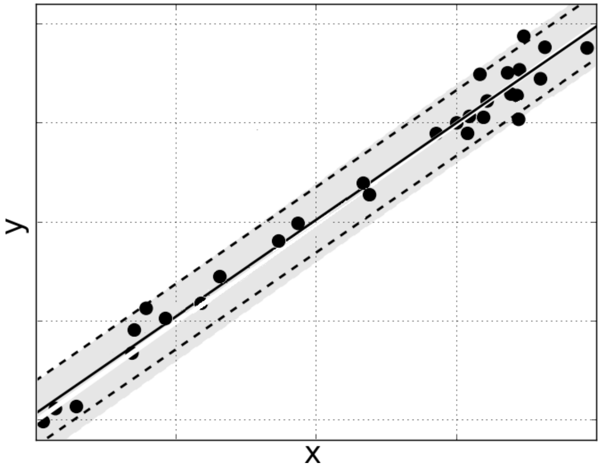
target values are assumed to be polluted by a Gaussian noise $\mathcal{N}\left(0, \sigma_t^2\right)$

$$\begin{aligned}
\log p\left(\mathbf{t}|\mathbf{X}, \mathbf{w}, \sigma_t\right) &= \sum_{i=1}^{n} \log p\left(t_i|\mathbf{x_i}, \mathbf{w}, \sigma_t\right) \\
&= \sum_{i=1}^{n} \log \mathcal{N}\left(t_i - f(x_{i1}, \ldots, x_{id})|0, \sigma_t^2\right) \\
&= \sum_{i=1}^{n} \log \frac{1}{\sqrt{2\pi\sigma_t^2}} \exp\left(-\frac{\left(t_i - f(x_{i1}, \ldots, x_{id})\right)^2}{2\sigma_t^2}\right) \\
&= -\frac{1}{2\sigma_t^2} \sum_{i=1}^{n} \left(t_i - f(x_{i1}, \ldots, x_{id})\right)^2 - \frac{n}{2} \log 2\pi\sigma_t^2.
\end{aligned}$$

max. $p\left(\mathbf{t}|\mathbf{X}, \mathbf{w}, \sigma_t\right) \Leftrightarrow$ min. $\frac{1}{\sigma_t^2} \times \underbrace{\frac{1}{n} \sum_{i=1}^{n} \left(t_i - f(x_{i1}, \ldots, x_{id})\right)^2}_{MSE} + \underbrace{\log \sigma_t^2}_{\sigma_t^2 \text{ term}}$

Maximum Likelihood Solution for Linear Regression

$$\mathbf{w} = \left(\mathbf{X}^T\mathbf{X}\right)^{-1}\mathbf{X}^T\mathbf{y} \qquad \sigma_t^2 = \frac{1}{n}\sum_{i=1}^{n}\left(t_i - f(x_{i1}, \ldots, x_{id})\right)^2$$



Classification with Linear Models

Why not Use Linear Regression for Classification?

Pros

- ✓ classes can be easily converted to numbers (e.g. 1, 2...)
- ✓ technically, nothing prevents us from using linear regression
- ✓ linear regression is very efficient and easy to understand

Cons

- × does not really make sense: targets are classes, not real numbers
- × objective function is not suitable (MSE of classes? really?)
- × results will change if we change class ordering (e.g. 1 2 3 $\Rightarrow$ 1 3 2)
- × classification is a very different problem calling for specific techniques (e.g. unbalanced classes, label noise, misclassification costs, etc.)

Why not Use Linear Regression for Classification?

Pros

- ✓ classes can be easily converted to numbers (e.g. 1, 2...)
- ✓ technically, nothing prevents us from using linear regression
- ✓ linear regression is very efficient and easy to understand

Cons

- × does not really make sense: targets are classes, not real numbers
- × objective function is not suitable (MSE of classes? really?)
- × results will change if we change class ordering (e.g. $1\,2\,3 \Rightarrow 1\,3\,2$)
- × classification is a very different problem calling for specific techniques (e.g. unbalanced classes, label noise, misclassification costs, etc.)

Logistic Regression

Linearity of the log-ratio

the log-ratio of the conditional class probabilities is assumed to be linear

$$\log\left(\frac{p(t=+1|\mathbf{x})}{p(t=-1|\mathbf{x})}\right) = w_1 x_1 + \cdots + w_d x_d + w_0$$

each weight w_j models the contribution of feature x_j to the log-ratio

- if $w_i x_i$ increases by 0.69, $\frac{p(t=+1|\mathbf{x})}{p(t=-1|\mathbf{x})}$ is doubled ($\exp 0.69 = 2$)

When do we use logistic regression ?

- normal class distributions with equal covariance matrices
- when number of features is large (too many to consider cross effects)
- when you think that only a few features are relevant $\Rightarrow$ selection
- typical applications: text mining, genetic data analysis...

Logistic Regression

Linearity of the log-ratio

the log-ratio of the conditional class probabilities is assumed to be linear

$$\log\left(\frac{p(t=+1|\mathbf{x})}{p(t=-1|\mathbf{x})}\right) = w_1x_1 + \cdots + w_dx_d + w_0$$

each weight w_j models the contribution of feature x_j to the log-ratio

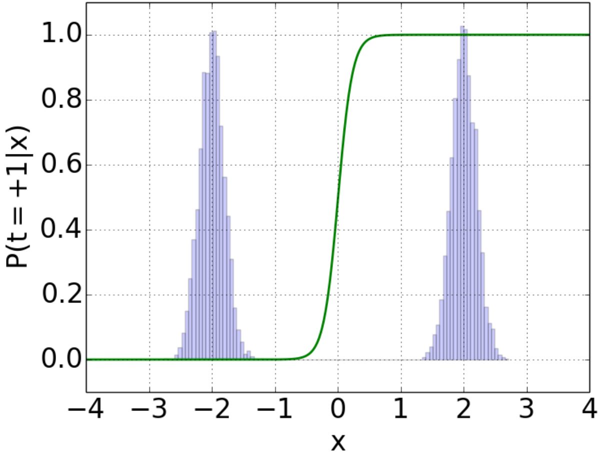
- if w_ix_i increases by 0.69, $\frac{p(t=+1|\mathbf{x})}{p(t=-1|\mathbf{x})}$ is doubled ($\exp 0.69 = 2$)

When do we use logistic regression ?

- normal class distributions with equal covariance matrices
- when number of features is large (too many to consider cross effects)
- when you think that only a few features are relevant $\Rightarrow$ selection
- typical applications: text mining, genetic data analysis...

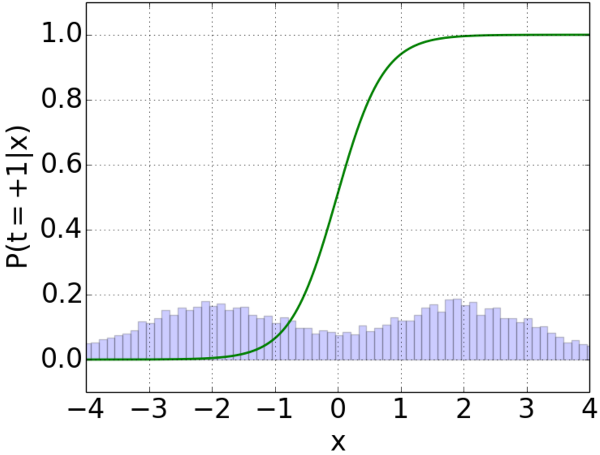
From log-ratio to conditional class probabilities

$$p(t=+1|\mathbf{x}) = \frac{1}{1+\exp\left(-\mathbf{w}^T\mathbf{x} - w_0\right)}$$



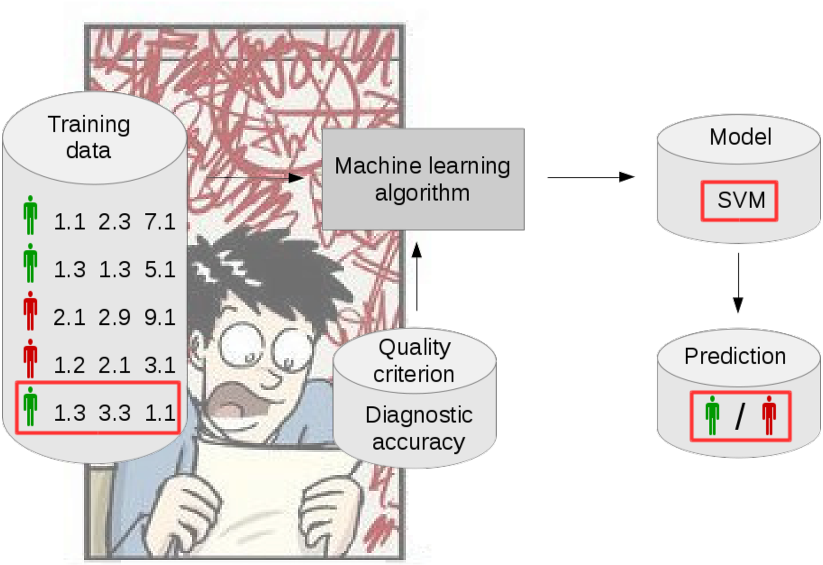
From log-ratio to conditional class probabilities

$$p(t = +1|\mathbf{x}) = \frac{1}{1 + \exp\left(-\mathbf{w}^T\mathbf{x} - w_0\right)}$$

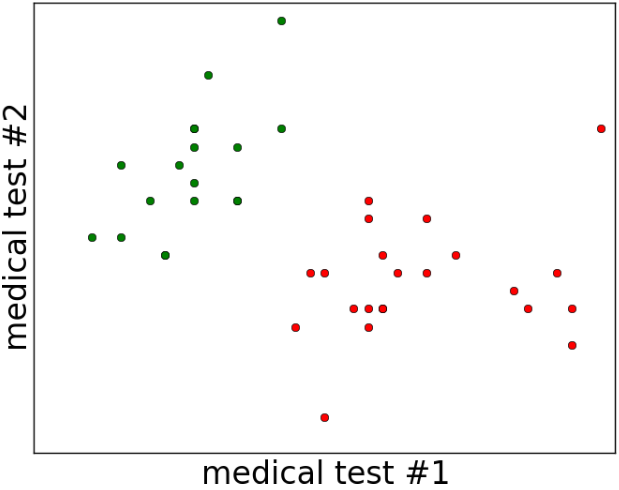


Outliers in Regression and Classification

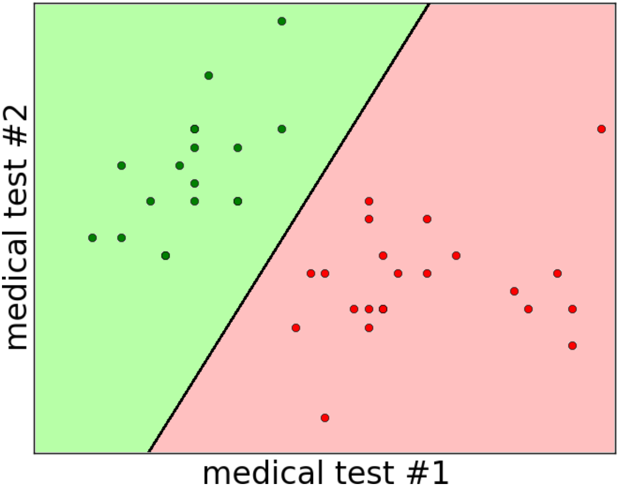
Challenges in Machine Learning: Robust Inference



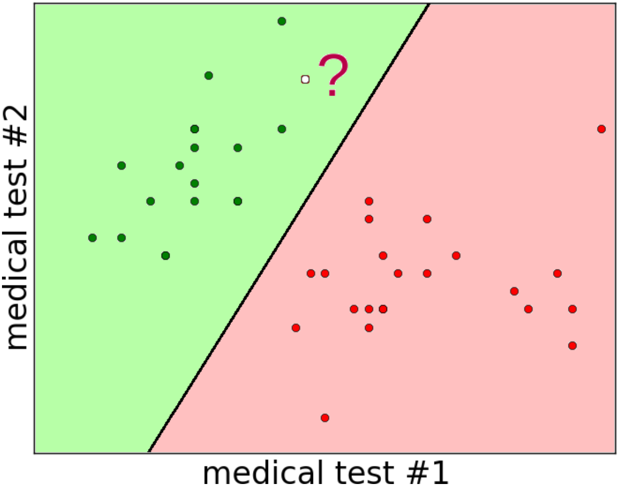
Classification with Clean Labels



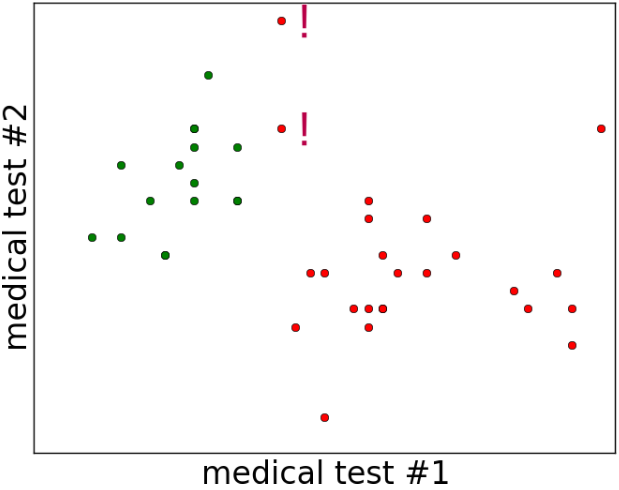
Classification with Clean Labels



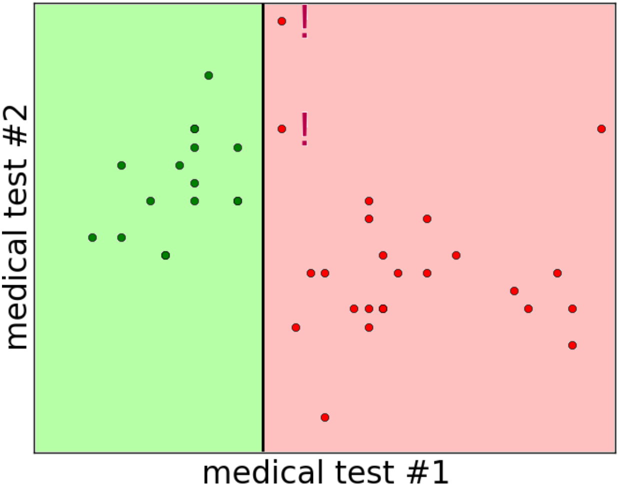
Classification with Clean Labels



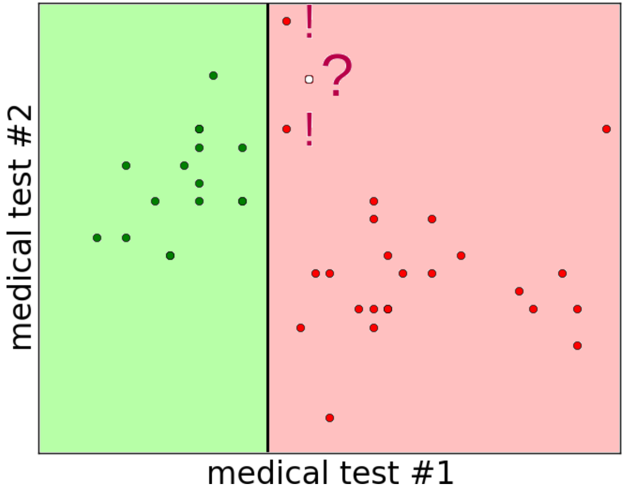
Classification with Label Noise



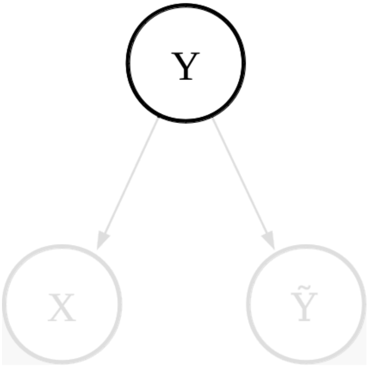
Classification with Label Noise



Classification with Label Noise



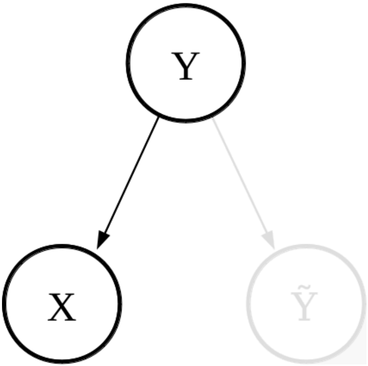
What is Label Noise (Probabilistic View)



Simple probabilistic model (binary classification)

$$P(\tilde{Y} = \tilde{y} | Y = y) = \begin{cases} .9 & \text{if } \tilde{y} = y\text{, i.e. there is } \textbf{no mislabelling} \\ .1 & \text{if } \tilde{y} \neq y\text{, i.e. there the } \textbf{label} \text{ is } \textbf{incorrect} \end{cases}$$

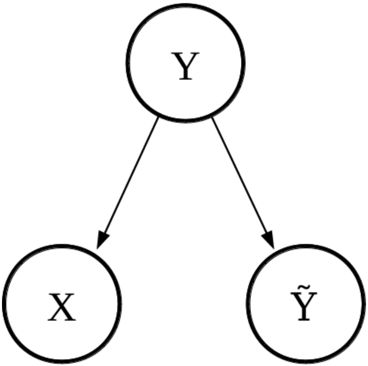
What is Label Noise (Probabilistic View)



Simple probabilistic model (binary classification)

$$P(\tilde{Y} = \tilde{y} | Y = y) = \begin{cases} .9 & \text{if } \tilde{y} = y \text{, i.e. there is } \textbf{no mislabelling} \\ .1 & \text{if } \tilde{y} \neq y \text{, i.e. there the } \textbf{label} \text{ is } \textbf{incorrect} \end{cases}$$

What is Label Noise (Probabilistic View)

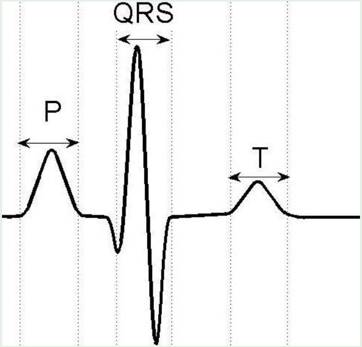


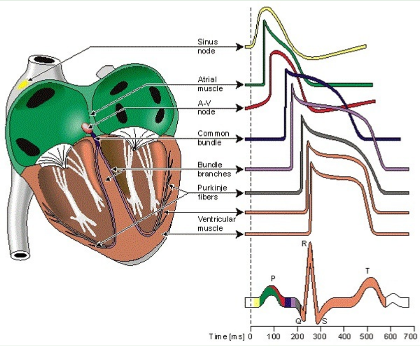
Simple probabilistic model (binary classification)

$$P(\tilde{Y} = \tilde{y} | Y = y) = \begin{cases} .9 & \text{if } \tilde{y} = y\text{, i.e. there is } \textbf{no mislabelling} \\ .1 & \text{if } \tilde{y} \neq y\text{, i.e. there the } \textbf{label} \text{ is } \textbf{incorrect} \end{cases}$$

Example of Label Noise: Electrocardiogram Signals

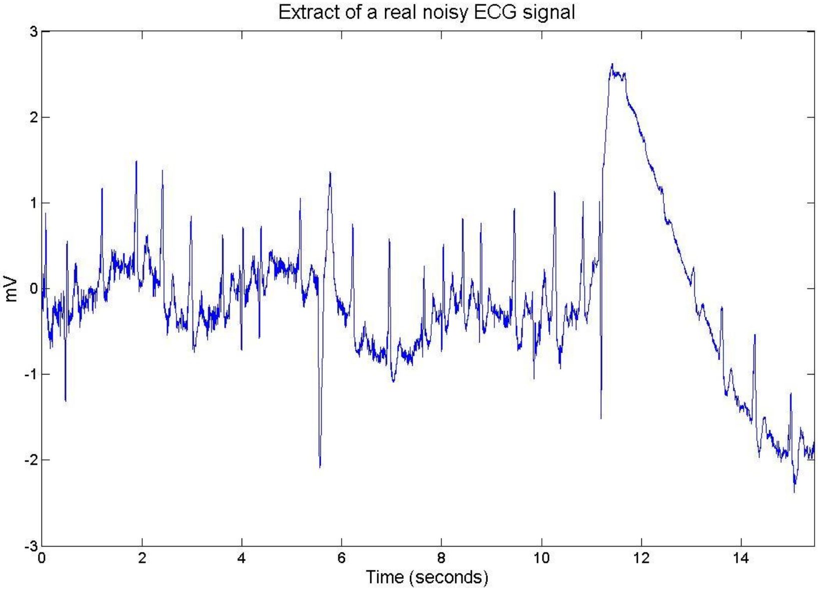
An ECG is a measure of the electrical activity of the human heart.



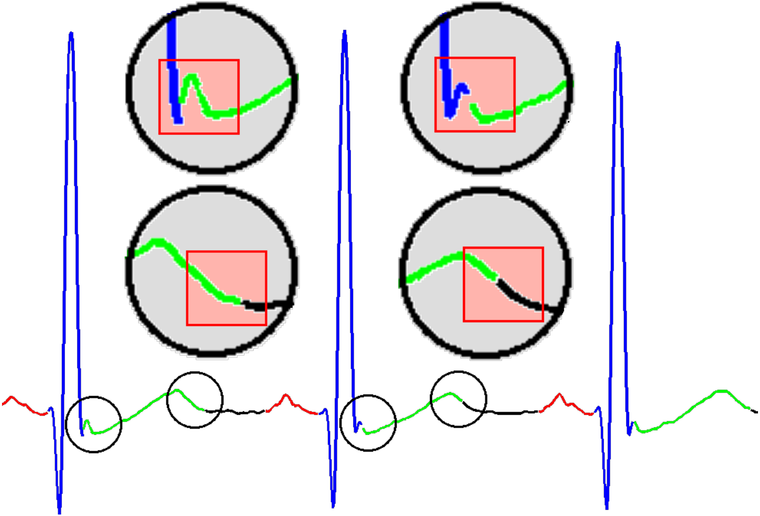


Patterns of interest: **P** wave, **QRS** complex, **T** wave, baseline.

Example of Label Noise: Electrocardiogram Signals



Example of Label Noise: Electrocardiogram Signals



Sources and Effects of Label Noise

Label noise can come from several sources

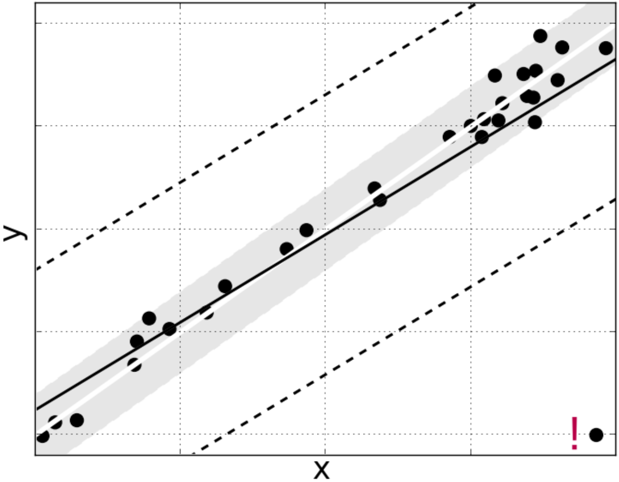
- insufficient information provided to the expert
- errors in the expert labelling itself
- subjectivity of the labelling task
- communication/encoding problems

Label noise can have several effects

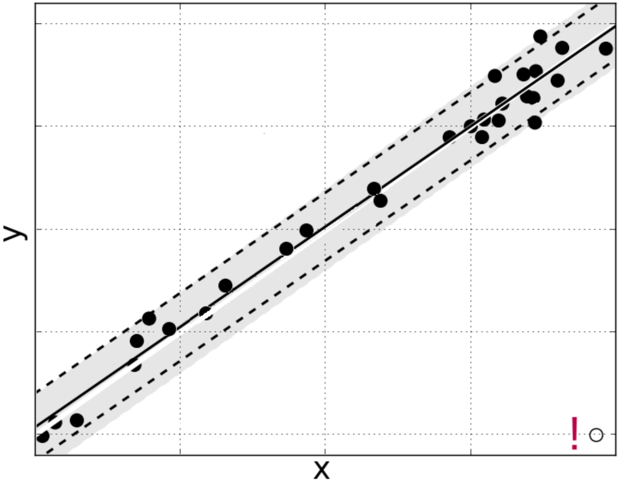
- decrease the classification performances
- increase the complexity of learned models
- pose a threat to tasks like e.g. feature selection

Source: Frénay, B., Verleysen, M. Classification in the Presence of Label Noise: a Survey. IEEE Trans. Neural Networks and Learning Systems.

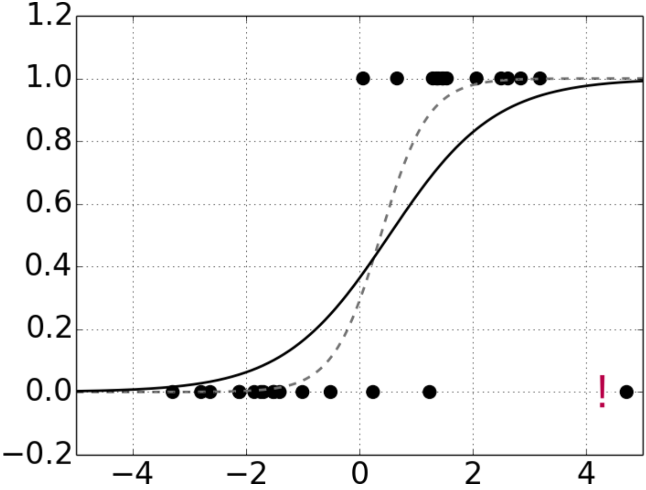
Robust Regression with Outliers



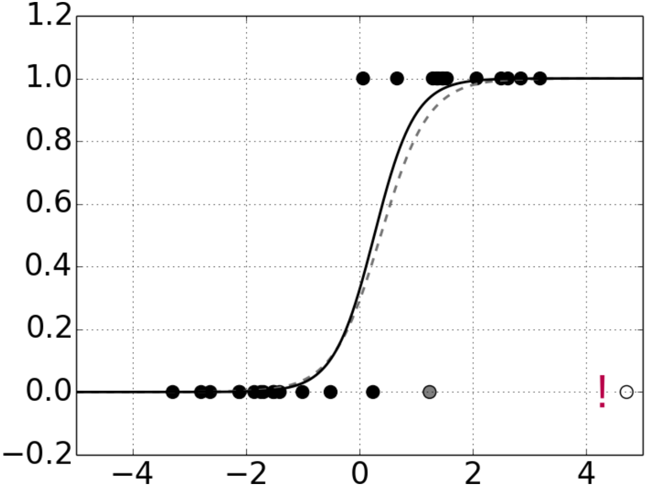
Robust Regression with Outliers



Robust Classification with Outliers



Robust Classification with Outliers



Outline of this Lesson

- regression with linear models
- classification with linear models
- outliers in regression and classification

References

robust inference: `https://bfrenay.wordpress.com/label-noise`