Machine Learning with Python

MTH786U/P 2023/24

Lecture 9: Interpreting regression models and logistic regression

Nicola Perra, Queen Mary University of London (QMUL)

n.perra@qmul.ac.uk

Regression models

In the previous lectures, we have studied regression problems of the form

$$\hat{\mathbf{w}} = \arg \min_{\mathbf{w} \in \mathbb{R}^{d+1}} E(\mathbf{w})$$

Regression models

In the previous lectures, we have studied regression problems of the form

$$\hat{\mathbf{w}} = \arg \min_{\mathbf{w} \in \mathbb{R}^{d+1}} E(\mathbf{w})$$

For

$$E(\mathbf{w}) = \mathrm{MSE}(\mathbf{w}) = \frac{1}{2s} \sum_{i=1}^{s} |f(x_i, w) - y_i|^2 \quad ,$$

where f is linear in w, we have seen that we can compute $\hat{w}$ by solving a linear system of equations

Regression models

In the previous lectures, we have studied regression problems of the form

$$\hat{\mathbf{w}} = \arg \min_{\mathbf{w} \in \mathbb{R}^{d+1}} E(\mathbf{w})$$

For

$$E(\mathbf{w}) = \frac{1}{2s} \sum_{i=1}^{s} |f(x_i, w) - y_i|^2 + \frac{\alpha}{2} \|\mathbf{w}\|^2 \quad ,$$

where f is linear in w, we have seen that we can compute $\hat{w}$ by solving a linear system of equations

Practical example in coursework6

Practical example in coursework6

The case of Boston houses

Practical example in coursework6

The case of Boston houses

For 1200 samples we have

- StreetLength - length of the street in front of the building
- Area - total area of the lot
- Quality - quality of building materials
- Condition - condition of the building
- BasementArea - area of the basement
- LivingArea - total living area
- GarageArea - a garage area
- SalePrice - sale price

Practical example in coursework6

The case of Boston houses

For 1200 samples we have

- StreetLength - length of the street in front of the building
- Area - total area of the lot
- Quality - quality of building materials
- Condition - condition of the building
- BasementArea - area of the basement
- LivingArea - total living area
- GarageArea - a garage area
- SalePrice - sale price ← target variable

Boston houses price regression case

We used K-fold cross validation to select the best value of the regularization term

Boston houses price regression case

We used K-fold cross validation to select the best value of the regularization term

We obtained this solution

Boston houses price regression case

We used K-fold cross validation to select the best value of the regularization term

We obtained this solution

```
An optimal value of regularisation parameter is 14.0.
For this value ofregularisation parameter one gets optimal weights of the form
[-0.00775026  0.06804823  0.45216932  0.03149377  0.14803397  0.26528746
  0.15823934]
```

Boston houses price regression case

We used K-fold cross validation to select the best value of the regularization term

We obtained this solution

```
An optimal value of regularisation parameter is 14.0.
For this value ofregularisation parameter one gets optimal weights of the form
[-0.00775026  0.06804823  0.45216932  0.03149377  0.14803397  0.26528746
  0.15823934]
```

This method is for model selection. In this case by model we consider: given a regression method (i.e., ridge regression) which is the best in terms of alpha

Boston houses price regression case

In our project you should/could try also different frameworks comparing their performance, say, using the MSE and always the k-fold cross validation for each

The output of this approach will be the best model among the ones you tried

Boston houses price regression case

To present the performance of the best model you can also use the coefficient of determination defined as the proportion of variation in the dependent variable explain by the independent variables

Boston houses price regression case

To present the performance of the best model you can also use the coefficient of determination defined as the proportion of variation in the dependent variable explain by the independent variables

$$R^2 = 1 - \frac{\sum_i^s (f_i - y_i)^2}{\sum_i^s (y_i - \langle y \rangle)^2} \qquad f_i = (\mathbf{X}\hat{\mathbf{W}})_i$$

Boston houses price regression case

Some of you by looking at

```
An optimal value of regularisation parameter is 14.0.
For this value ofregularisation parameter one gets optimal weights of the form
[-0.00775026  0.06804823  0.45216932  0.03149377  0.14803397  0.26528746
  0.15823934]
```

Might think, so what? What do we learn from this? How can we interpret the results?

Boston houses price regression case

Some of you by looking at

```
An optimal value of regularisation parameter is 14.0.
For this value ofregularisation parameter one gets optimal weights of the form
[-0.00775026  0.06804823  0.45216932  0.03149377  0.14803397  0.26528746
  0.15823934]
```

Might think, so what? What do we learn from this? How can we interpret the results?

Model selection and model interpretation are different goals!

Boston houses price regression case

In some cases you just care to get the best model as possible because you want to predict, for example, the price of a new house in the database -> use the model

Boston houses price regression case

In some cases you just care to get the best model as possible because you want to predict, for example, the price of a new house in the database -> use the model

In some cases you are more interested at interpreting the model's outcome. For example answering questions such as which is the most important feature?

Boston houses price regression case

How can we interpret the outcomes of a regression?

```
An optimal value of regularisation parameter is 14.0.
For this value ofregularisation parameter one gets optimal weights of the form
[-0.00775026  0.06804823  0.45216932  0.03149377  0.14803397  0.26528746
  0.15823934]
```

Boston houses price regression case

If you standardise the inputs/outputs each of these w_i can be interpreted as the variation in the output resulting from an increase of a standard deviation in that w_i

```
An optimal value of regularisation parameter is 14.0.
For this value ofregularisation parameter one gets optimal weights of the form
[-0.00775026  0.06804823  0.45216932  0.03149377  0.14803397  0.26528746
  0.15823934]
```

Boston houses price regression case

If you standardise the inputs/outputs each of these w_i can be interpreted as the variation in the output resulting from an increase of a standard deviation in that w_i

So, an increase of one standard deviation in w_2 (area) will result in a change in standardized price of 0.068, in w_3 (condition) of 0.45

```
An optimal value of regularisation parameter is 14.0.
For this value ofregularisation parameter one gets optimal weights of the form
[-0.00775026  0.06804823  0.45216932  0.03149377  0.14803397  0.26528746
  0.15823934]
```

Boston houses price regression case

How can we interpret the outcomes of a regression?

```
An optimal value of regularisation parameter is 14.0.
For this value ofregularisation parameter one gets optimal weights of the form
[-0.00775026  0.06804823  0.45216932  0.03149377  0.14803397  0.26528746
  0.15823934]
```

Boston houses price regression case

How can we interpret the outcomes of a regression?

```
An optimal value of regularisation parameter is 14.0.
For this value ofregularisation parameter one gets optimal weights of the form
[-0.00775026  0.06804823  0.45216932  0.03149377  0.14803397  0.26528746
  0.15823934]
```

We can order them

```
[0.45216932052319875, 'Quality']
[0.265287464036829, 'LivingArea']
[0.15823933619435454, 'GarageArea']
[0.14803396529403573, 'BasementArea']
[0.06804822890073225, 'Area']
[0.03149376548286966, 'Condition']
[-0.007750257643615537, 'StreetLength']
```

Boston houses price regression case

Little issue

```
An optimal value of regularisation parameter is 14.0.
For this value ofregularisation parameter one gets optimal weights of the form
[-0.00775026  0.06804823  0.45216932  0.03149377  0.14803397  0.26528746
  0.15823934]
```

Boston houses price regression case

Little issue

```
An optimal value of regularisation parameter is 14.0.
For this value ofregularisation parameter one gets optimal weights of the form
[-0.00775026  0.06804823  0.45216932  0.03149377  0.14803397  0.26528746
  0.15823934]
```

The output of this method gives us only one value of $\hat{\mathbf{w}}$

Boston houses price regression case

Little issue

```
An optimal value of regularisation parameter is 14.0.
For this value ofregularisation parameter one gets optimal weights of the form
[-0.00775026  0.06804823  0.45216932  0.03149377  0.14803397  0.26528746
  0.15823934]
```

The output of this method gives us only one value of $\hat{\mathbf{w}}$

How can we provide better estimates for their values, providing for example confidence intervals?

Boston houses price regression case

Little issue

```
An optimal value of regularisation parameter is 14.0.
For this value ofregularisation parameter one gets optimal weights of the form
[-0.00775026  0.06804823  0.45216932  0.03149377  0.14803397  0.26528746
  0.15823934]
```

The output of this method gives us only one value of $\hat{\mathbf{w}}$

How can we provide better estimates for their values, providing for example confidence intervals?

Bootstrap sampling!

Bootstrap sampling

The idea: we can sample the original data (with replacement) and create many instances of the data

Bootstrap sampling

The idea: we can sample the original data (with replacement) and create many instances of the data

In practice: pick at random (with replacement) N samples, that is, $(\mathbf{x}, y)$ repeat the extraction M times

Bootstrap sampling

The idea: we can sample the original data (with replacement) and create many instances of the data

In practice: pick at random (with replacement) N samples, that is, $(\mathbf{x}, y)$ repeat the extraction M times

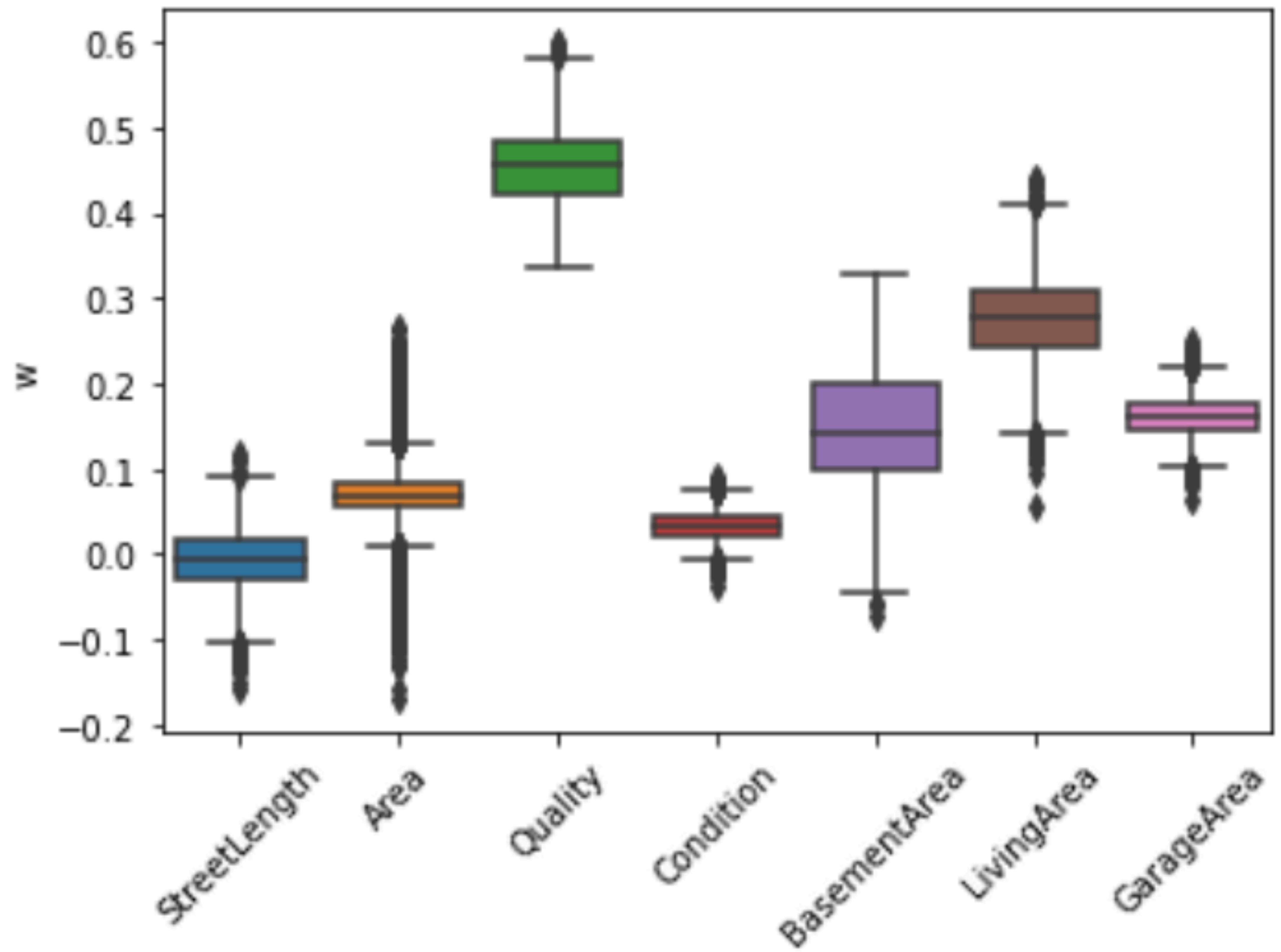
For each of the M samples, we can do a regression, get $\hat{\mathbf{w}}$
then considering the M instances we can compute estimates of them!

Bootstrap sampling

```python
import random as rd
def bootstrap_regression(standardised_data_input,standardised_data_output,fraction,M,alpha):
    # first we need to know what is N: the number of samples to extract
    data_size=len(standardised_data_output)
    samples_size=int(data_size*fraction)

    w_list=[]
    # then for each of the M extract
    for j in range(M):
        sample_input_list=[]
        sample_output_list=[]
        for i in range(samples_size):
            # we take N samples extract random numbers which are the id of the arrays that store the data
            id_random=rd.randint(0,samples_size-1)
            # note that we need to keep the X and Y correspondence hence the id_random is the same for each
            sample_input_list.append(standardised_data_input[id_random])
            sample_output_list.append(standardised_data_output[id_random])

        # convert the list to arrays
        sample_input=np.array(sample_input_list)
        sample_output=np.array(sample_output_list)

        # apply the regression, note that alpha is selected before with the model selection
        weights=ridge_regression(sample_input, sample_output, regularisation=alpha)
        # append the fitted values of Ws for the N samples in list
        w_list.append(weights)
    return w_list
```

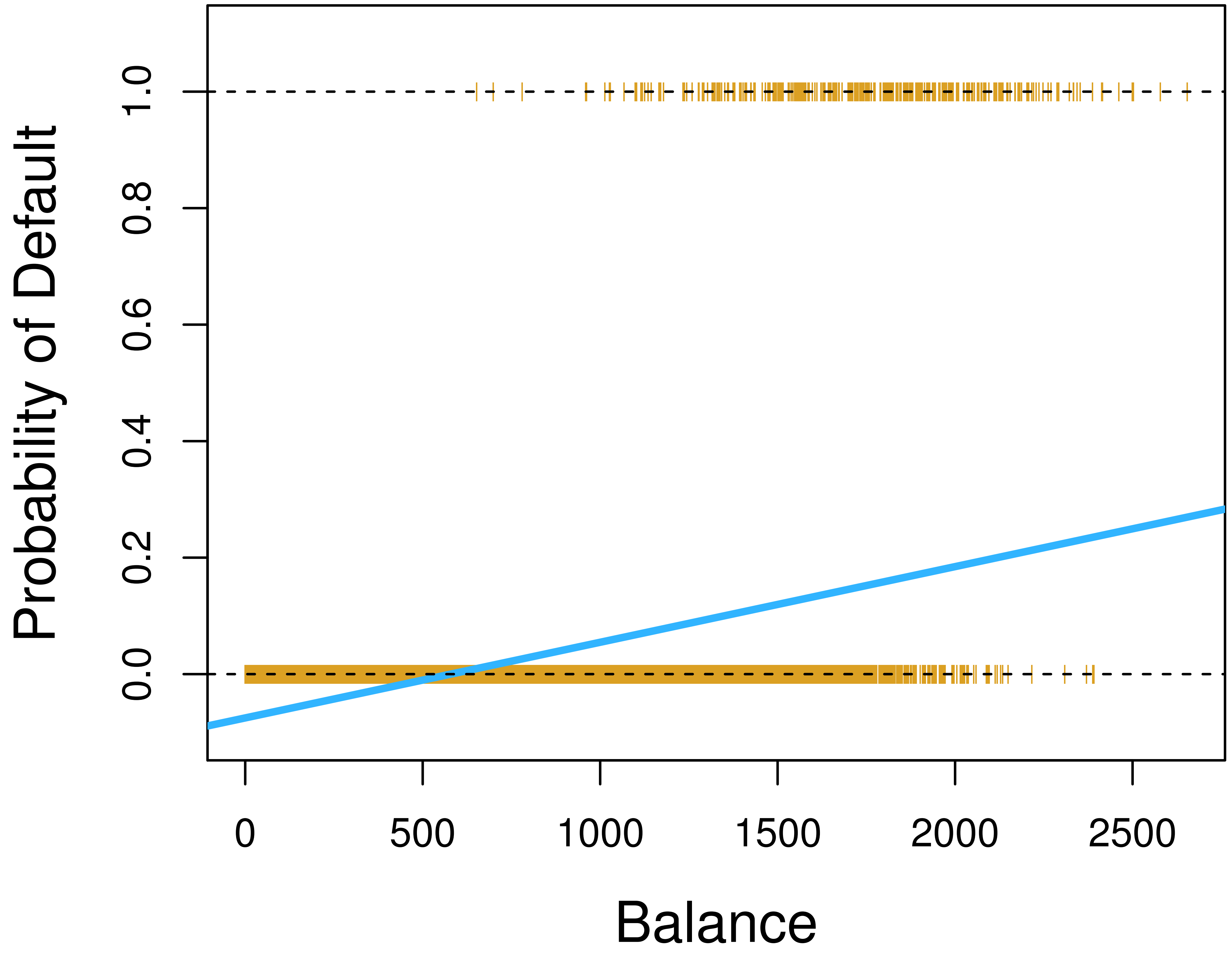
Bootstrap sampling

Results using M=10000 samples



Logistic regression

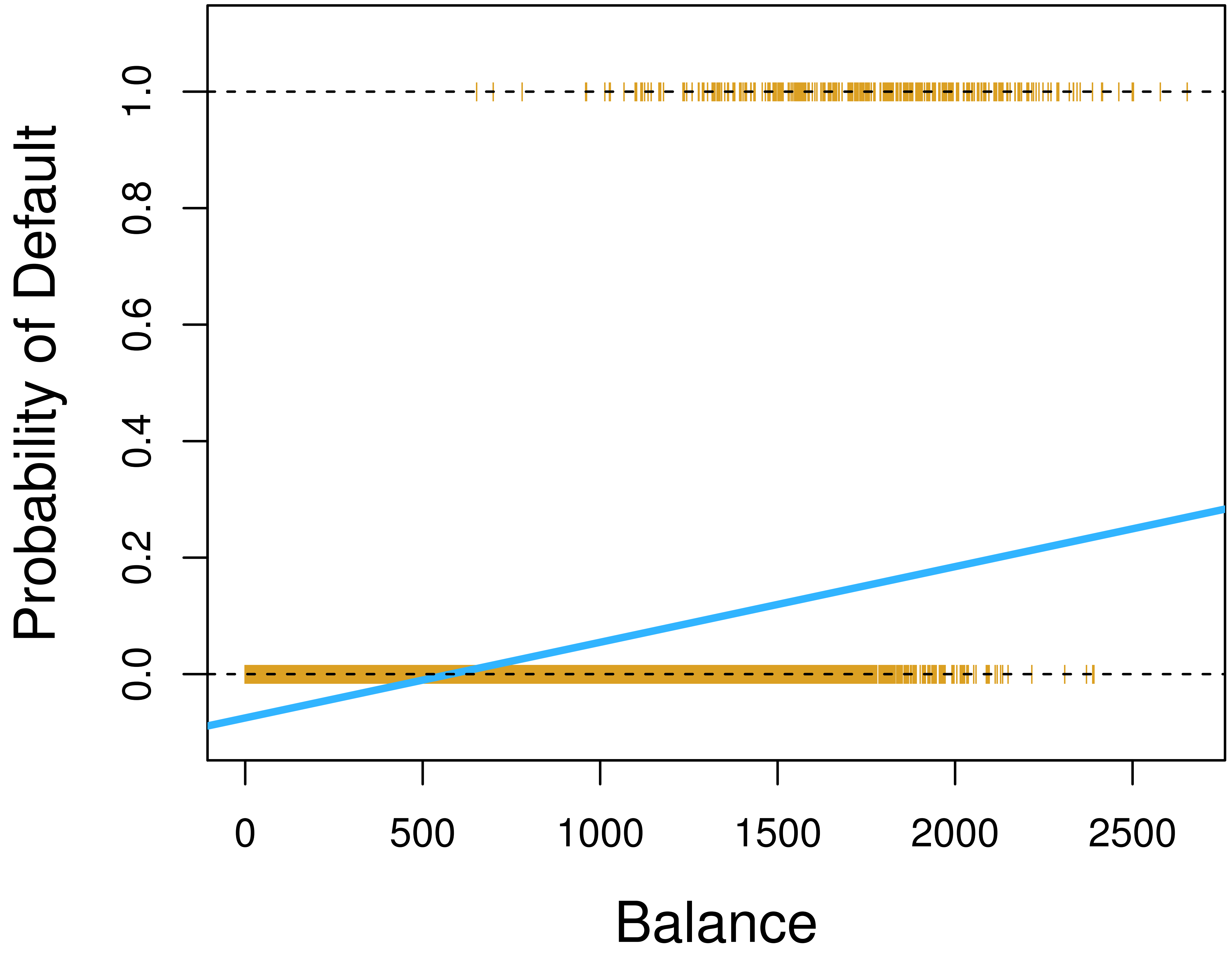
Issues with MSE regression for classification:



Logistic regression

Issues with MSE regression for classification:

Predicted values are usually not in $[0,1]$

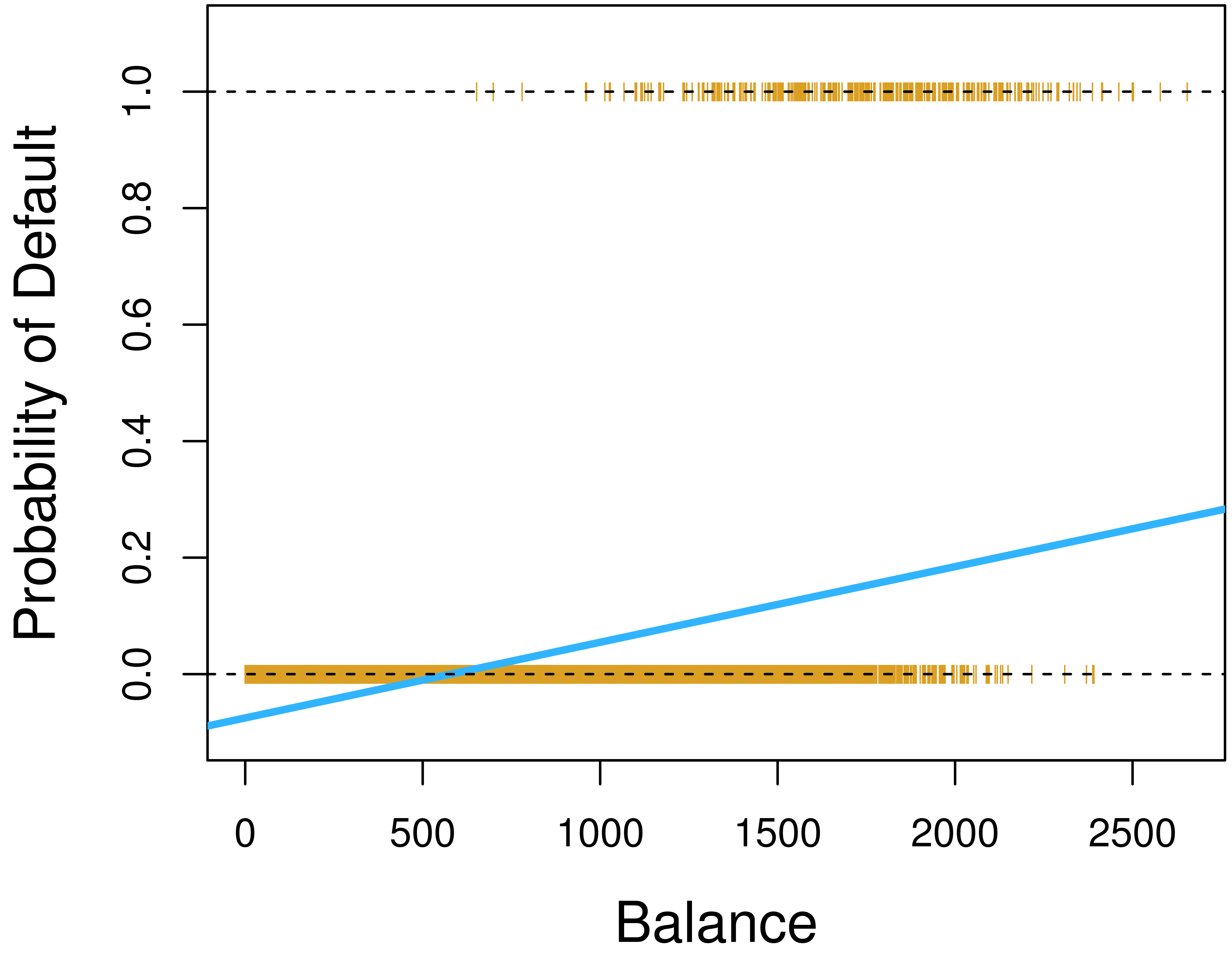


Logistic regression

Issues with MSE regression for classification:

Predicted values are usually not in $[0,1]$

If the predicted values would be much smaller than zero or larger than one, the MSE would penalize them though they would be very confident output of the classification



Logistic regression

It seems reasonable to transform the prediction into a probability, i.e.

$$\text{consider} \quad \sigma(\langle \mathbf{x}_i, \mathbf{w} \rangle) \quad \text{instead of} \quad \langle \mathbf{x}_i, \mathbf{w} \rangle$$

with $\sigma : (-\infty, \infty) \rightarrow [0,1]$

Logistic regression

It seems reasonable to transform the prediction into a probability, i.e.

consider $\sigma(\langle \mathbf{x}_i, \mathbf{w} \rangle)$ instead of $\langle \mathbf{x}_i, \mathbf{w} \rangle$

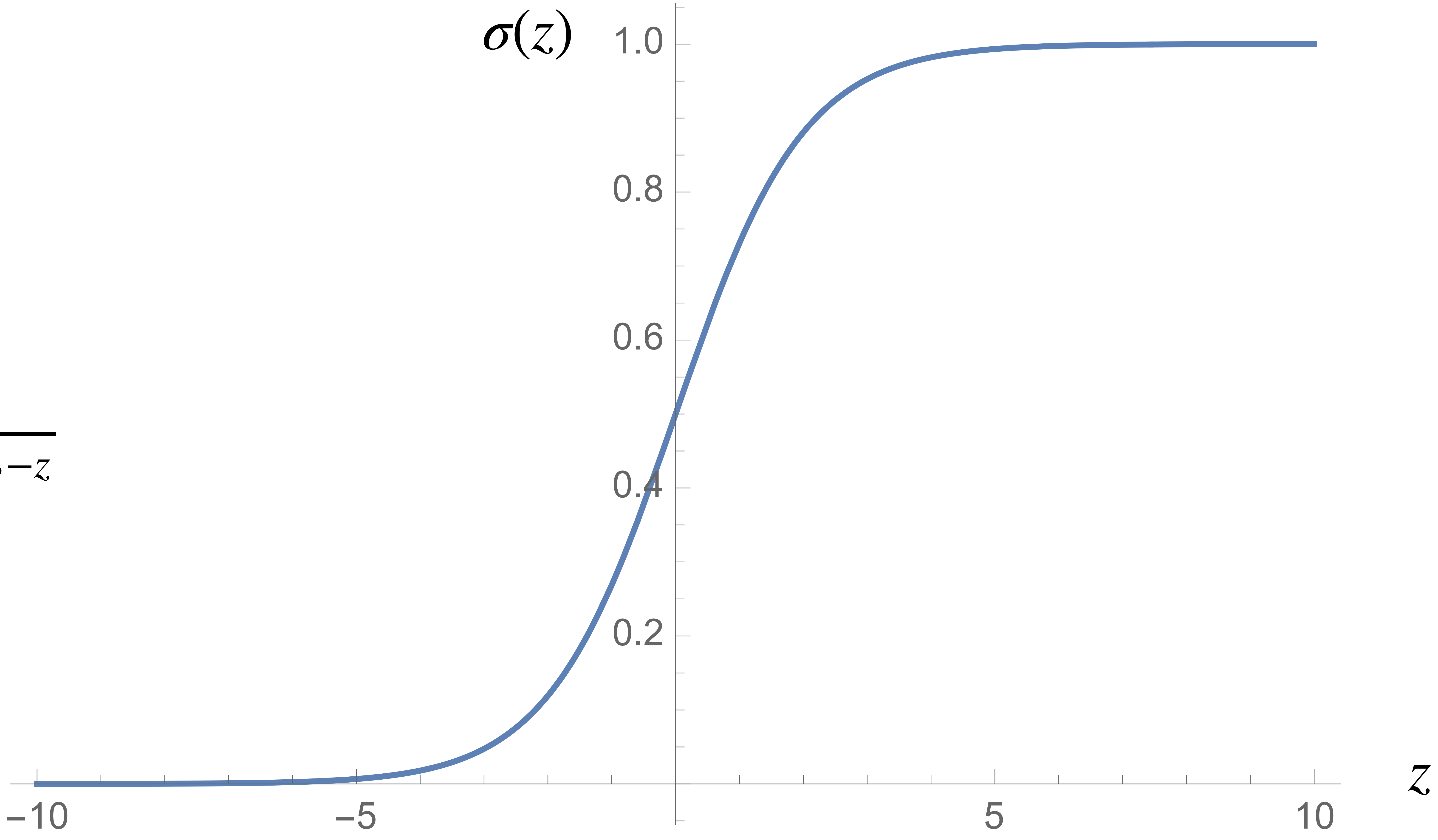
with $\sigma : (-\infty, \infty) \to [0,1]$

Many ways of doing so; popular choice is the logistic function

$$\sigma(z) := \frac{e^z}{1 + e^z}$$

Logistic regression

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



Logistic regression

Consider binary classification with class labels 0 and 1

Logistic regression

Consider binary classification with class labels 0 and 1

Input/output training samples $\{(\mathbf{x}_i, y_i)\}_{i=1}^{s}$ with $y_i \in \{0,1\}$

Logistic regression

Consider binary classification with class labels 0 and 1

Input/output training samples $\{(\mathbf{x}_i, y_i)\}_{i=1}^{s}$ with $y_i \in \{0,1\}$

Model assumption: $f(\mathbf{x}_i, \mathbf{w}) = \langle \mathbf{x}_i, \mathbf{w} \rangle$

Logistic regression

Consider binary classification with class labels 0 and 1

Input/output training samples $\{(\mathbf{x}_i, y_i)\}_{i=1}^{s}$ with $y_i \in \{0,1\}$

Model assumption: $f(\mathbf{x}_i, \mathbf{w}) = \langle \mathbf{x}_i, \mathbf{w} \rangle$

Posterior probability of the two class labels given $\mathbf{x}$ is:

$$\rho(1 \,|\, \mathbf{x}) = \sigma(\langle \mathbf{x}, \mathbf{w} \rangle)$$

$$\rho(0 \,|\, \mathbf{x}) = 1 - \sigma(\langle \mathbf{x}, \mathbf{w} \rangle)$$

Logistic regression

Training: how do we obtain optimal parameters $\hat{\mathbf{w}}$ given input/output samples $\{(\mathbf{x}_i, y_i)\}_{i=1}^{s}$?

Logistic regression

Training: how do we obtain optimal parameters $\hat{\mathbf{w}}$ given input/output samples $\{(\mathbf{x}_i, y_i)\}_{i=1}^{s}$?

Assumption (as always): samples $(\mathbf{x}_i, y_i)$ are iid

Logistic regression

Training: how do we obtain optimal parameters $\hat{\mathbf{w}}$ given input/output samples $\{(\mathbf{x}_i, y_i)\}_{i=1}^s$?

Assumption (as always): samples $(\mathbf{x}_i, y_i)$ are iid

Then the likelihood of y given X and w is $$\rho(\mathbf{y} \,|\, \mathbf{X}, \mathbf{w}) = \prod_{i=1}^{s} \rho(y_i \,|\, \mathbf{x}_i)$$

for $\mathbf{y} := \begin{pmatrix} y_1 \\ y_2 \\ \vdots \\ y_s \end{pmatrix}$, $\mathbf{X} := \begin{pmatrix} \mathbf{x}_1^\top \\ \mathbf{x}_2^\top \\ \vdots \\ \mathbf{x}_s^\top \end{pmatrix}$ and $\mathbf{x}_i := \begin{pmatrix} 1 \\ x_{i1} \\ x_{i2} \\ \vdots \\ x_{id} \end{pmatrix}$

Logistic regression

$$\rho(\mathbf{y} \mid \mathbf{X}, \mathbf{w}) = \prod_{i=1}^{s} \rho(y_i \mid \mathbf{x}_i)$$

Logistic regression

$$\rho(\mathbf{y} \mid \mathbf{X}, \mathbf{w}) = \prod_{i=1}^{s} \rho(y_i \mid \mathbf{x}_i)$$

$$= \prod_{i=1}^{s} \sigma(\langle \mathbf{x}_i, \mathbf{w} \rangle)^{y_i} (1 - \sigma(\langle \mathbf{x}_i, \mathbf{w} \rangle))^{1-y_i}$$

Logistic regression

$$\rho(\mathbf{y} \mid \mathbf{X}, \mathbf{w}) = \prod_{i=1}^{s} \rho(y_i \mid \mathbf{x}_i)$$

$$= \prod_{i=1}^{s} \sigma(\langle \mathbf{x}_i, \mathbf{w} \rangle)^{y_i} (1 - \sigma(\langle \mathbf{x}_i, \mathbf{w} \rangle))^{1-y_i}$$

Negative log-likelihood:

$$-\log(\, \rho(\mathbf{y} \mid \mathbf{X}, \mathbf{w})\,) = -\log \left(\prod_{i=1}^{s} \sigma(\langle \mathbf{x}_i, \mathbf{w} \rangle)^{y_i} (1 - \sigma(\langle \mathbf{x}_i, \mathbf{w} \rangle))^{1-y_i} \right)$$

Logistic regression

Negative log-likelihood:

$$-\log(\,\rho(\mathbf{y}\,|\,\mathbf{X},\mathbf{w})\,) = -\log\left(\prod_{i=1}^{s}\sigma(\langle\mathbf{x}_i,\mathbf{w}\rangle)^{y_i}(1-\sigma(\langle\mathbf{x}_i,\mathbf{w}\rangle))^{1-y_i}\right)$$

$$= -\sum_{i=1}^{s}\left[y_i\log(\sigma(\langle\mathbf{x}_i,\mathbf{w}\rangle)) + (1-y_i)\log(1-\sigma(\langle\mathbf{x}_i,\mathbf{w}\rangle))\right]$$

Logistic regression

Negative log-likelihood:

$$-\log(\,\rho(\mathbf{y}\,|\,\mathbf{X},\mathbf{w})\,) = -\log\left(\prod_{i=1}^{s}\sigma(\langle\mathbf{x}_i,\mathbf{w}\rangle)^{y_i}\,(1-\sigma(\langle\mathbf{x}_i,\mathbf{w}\rangle))^{1-y_i}\right)$$

$$= -\sum_{i=1}^{s}\left[y_i\log(\sigma(\langle\mathbf{x}_i,\mathbf{w}\rangle)) + (1-y_i)\log(1-\sigma(\langle\mathbf{x}_i,\mathbf{w}\rangle))\right]$$

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

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

Logistic regression

Negative log-likelihood:

$$-\log(\,\rho(\mathbf{y}\,|\,\mathbf{X},\mathbf{w})\,) = -\sum_{i=1}^{s}\left[y_i\log(\sigma(\langle\mathbf{x}_i,\mathbf{w}\rangle)) + (1-y_i)\log(1-\sigma(\langle\mathbf{x}_i,\mathbf{w}\rangle))\right]$$

$$= -\sum_{i=1}^{s}\left[y_i\log\left(\frac{e^{\langle\mathbf{x}_i,\mathbf{w}\rangle}}{1+e^{\langle\mathbf{x}_i,\mathbf{w}\rangle)}}\right) + (1-y_i)\log\left(\frac{1}{1+e^{\langle\mathbf{x}_i,\mathbf{w}\rangle}}\right)\right]$$

Logistic regression

Negative log-likelihood:

$$-\log(\,\rho(\mathbf{y}\,|\,\mathbf{X}, \mathbf{w})\,) = -\sum_{i=1}^{s}\left[y_i \log(\sigma(\langle \mathbf{x}_i, \mathbf{w}\rangle)) + (1 - y_i)\log(1 - \sigma(\langle \mathbf{x}_i, \mathbf{w}\rangle))\right]$$

$$= -\sum_{i=1}^{s}\left[y_i \log\left(\frac{e^{\langle \mathbf{x}_i, \mathbf{w}\rangle}}{1 + e^{\langle \mathbf{x}_i, \mathbf{w}\rangle)}}\right) + (1 - y_i)\log\left(\frac{1}{1 + e^{\langle \mathbf{x}_i, \mathbf{w}\rangle}}\right)\right]$$

$$= \sum_{i=1}^{s} \log\left(1 + \exp(\langle \mathbf{x}_i, \mathbf{w}\rangle)\right) - y_i \langle \mathbf{x}_i, \mathbf{w}\rangle$$

Logistic regression

$$-\log(\,\rho(\mathbf{y}\,|\,\mathbf{X},\mathbf{w})\,) = \sum_{i=1}^{s} \log\left(1 + \exp(\langle \mathbf{x}_i, \mathbf{w}\rangle)\right) - y_i\,\langle \mathbf{x}_i, \mathbf{w}\rangle$$

Logistic regression

$$-\log(\rho(\mathbf{y} \,|\, \mathbf{X}, \mathbf{w})) = \sum_{i=1}^{s} \log\left(1 + \exp(\langle \mathbf{x}_i, \mathbf{w} \rangle)\right) - y_i \langle \mathbf{x}_i, \mathbf{w} \rangle$$

$$\hat{\mathbf{w}} = \arg\min_{w} \left\{ -\log(\rho(\mathbf{y} \,|\, \mathbf{X}, \mathbf{w})) \right\}$$

Logistic regression

$$-\log(\rho(\mathbf{y} \mid \mathbf{X}, \mathbf{w})) = \sum_{i=1}^{s} \log\left(1 + \exp(\langle \mathbf{x}_i, \mathbf{w} \rangle)\right) - y_i \langle \mathbf{x}_i, \mathbf{w} \rangle$$

$$\hat{\mathbf{w}} = \arg\min_{w} \left\{ -\log(\rho(\mathbf{y} \mid \mathbf{X}, \mathbf{w})) \right\}$$

$$= \arg\min_{w} \left\{ \sum_{i=1}^{s} \log\left(1 + \exp(\langle \mathbf{x}_i, \mathbf{w} \rangle)\right) - y_i \langle \mathbf{x}_i, \mathbf{w} \rangle \right\}$$

Logistic regression

$$-\log(\rho(\mathbf{y} \,|\, \mathbf{X}, \mathbf{w})) = \sum_{i=1}^{s} \log\left(1 + \exp(\langle \mathbf{x}_i, \mathbf{w} \rangle)\right) - y_i \langle \mathbf{x}_i, \mathbf{w} \rangle$$

$$\hat{\mathbf{w}} = \arg\min_{w} \left\{ -\log(\rho(\mathbf{y} \,|\, \mathbf{X}, \mathbf{w})) \right\}$$

$$= \arg\min_{w} \left\{ \sum_{i=1}^{s} \log\left(1 + \exp(\langle \mathbf{x}_i, \mathbf{w} \rangle)\right) - y_i \langle \mathbf{x}_i, \mathbf{w} \rangle \right\}$$

$\Rightarrow$ $\hat{\mathbf{w}}$ maximises the likelihood

(i.e. maximises the probability of observing $\mathbf{y}$, given $\mathbf{X}$)

Multinomial logistic regression

The key idea is to model multiple classes with a probability simplex

(~ discrete probability density)

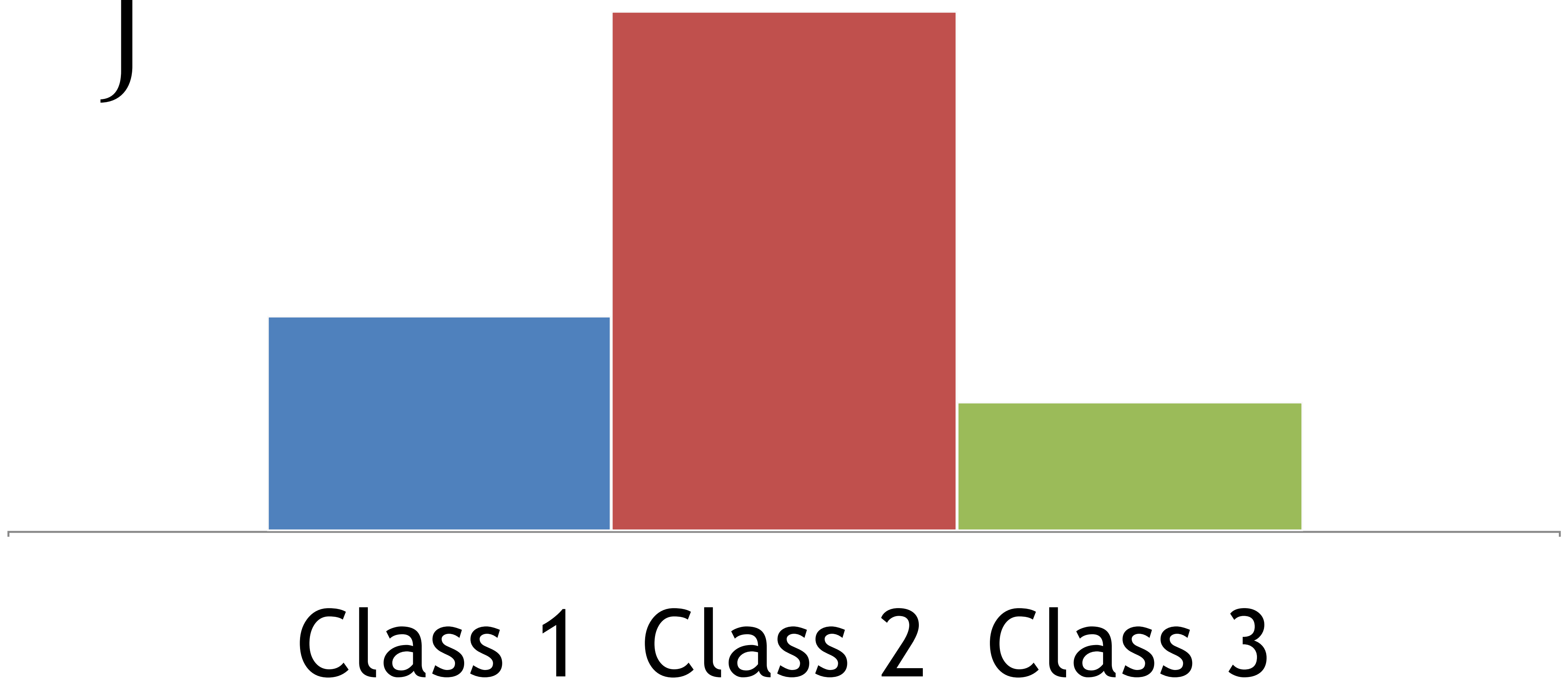
$$\Sigma := \left\{ \rho \in \mathbb{R}^n \;\middle|\; \rho_i \geq 0 \text{ for } i \in \{1,\ldots,n\} \text{ and } \sum_{i=1}^{n} \rho_i = 1 \right\}$$

Multinomial logistic regression

The key idea is to model multiple classes with a probability simplex

(~ discrete probability density)

$$\Sigma := \left\{ \rho \in \mathbb{R}^n \;\middle|\; \rho_i \geq 0 \text{ for } i \in \{1, \ldots, n\} \text{ and } \sum_{i=1}^{n} \rho_i = 1 \right\}$$



Multinomial logistic regression

How can we map a vector onto the probability simplex?

Multinomial logistic regression

How can we map a vector onto the probability simplex?

One way to do it: *via* the softmax function

$$\sigma(\mathbf{v}) = \mathsf{softmax}(\mathbf{v}) := \left(\frac{\exp(v_1)}{\sum_{j=1}^{n} \exp(v_j)} \quad \frac{\exp(v_2)}{\sum_{j=1}^{n} \exp(v_j)} \quad \cdots \quad \frac{\exp(v_n)}{\sum_{j=1}^{n} \exp(v_j)} \right)$$

Multinomial logistic regression

How can we map a vector onto the probability simplex?

One way to do it: *via* the softmax function

$$\sigma(\mathbf{v}) = \text{softmax}(\mathbf{v}) := \left(\frac{\exp(v_1)}{\sum_{j=1}^{n} \exp(v_j)} \quad \frac{\exp(v_2)}{\sum_{j=1}^{n} \exp(v_j)} \quad \cdots \quad \frac{\exp(v_n)}{\sum_{j=1}^{n} \exp(v_j)} \right)$$

Component-wise:

$$\sigma(\mathbf{v})_i = \frac{\exp(v_i)}{\sum_{j=1}^{n} \exp(v_j)}$$

Multinomial logistic regression

How can we map a vector onto the probability simplex?

$$\sigma(\mathbf{v}) = \text{softmax}(\mathbf{v}) := \left(\frac{\exp(v_1)}{\sum_{j=1}^{n} \exp(v_j)} \quad \frac{\exp(v_2)}{\sum_{j=1}^{n} \exp(v_j)} \quad \cdots \quad \frac{\exp(v_n)}{\sum_{j=1}^{n} \exp(v_j)} \right)$$

Why is it called softmax?

Multinomial logistic regression

How can we map a vector onto the probability simplex?

$$\sigma(\mathbf{v}) = \text{softmax}(\mathbf{v}) := \begin{pmatrix} \frac{\exp(v_1)}{\sum_{j=1}^{n} \exp(v_j)} & \frac{\exp(v_2)}{\sum_{j=1}^{n} \exp(v_j)} & \cdots & \frac{\exp(v_n)}{\sum_{j=1}^{n} \exp(v_j)} \end{pmatrix}$$

Why is it called softmax?

Example: $\begin{pmatrix} 1.5 & 0.3 & -3.7 \end{pmatrix}$

Multinomial logistic regression

How can we map a vector onto the probability simplex?

$$\sigma(\mathbf{v}) = \text{softmax}(\mathbf{v}) := \begin{pmatrix} \frac{\exp(v_1)}{\sum_{j=1}^n \exp(v_j)} & \frac{\exp(v_2)}{\sum_{j=1}^n \exp(v_j)} & \cdots & \frac{\exp(v_n)}{\sum_{j=1}^n \exp(v_j)} \end{pmatrix}$$

Why is it called softmax?

Example: $\begin{pmatrix} 1.5 & 0.3 & -3.7 \end{pmatrix}$

$$\arg\max \begin{pmatrix} 1.5 & 0.3 & -3.7 \end{pmatrix} = 0$$

Multinomial logistic regression

How can we map a vector onto the probability simplex?

$$\sigma(\mathbf{v}) = \text{softmax}(\mathbf{v}) := \begin{pmatrix} \frac{\exp(v_1)}{\sum_{j=1}^{n} \exp(v_j)} & \frac{\exp(v_2)}{\sum_{j=1}^{n} \exp(v_j)} & \cdots & \frac{\exp(v_n)}{\sum_{j=1}^{n} \exp(v_j)} \end{pmatrix}$$

Why is it called softmax?

Example: $\begin{pmatrix} 1.5 & 0.3 & -3.7 \end{pmatrix}$

$$\arg\max \begin{pmatrix} 1.5 & 0.3 & -3.7 \end{pmatrix} = 0$$

The max argument of the vector is the value in position 0

Multinomial logistic regression

How can we map a vector onto the probability simplex?

$$\sigma(\mathbf{v}) = \text{softmax}(\mathbf{v}) := \begin{pmatrix} \frac{\exp(v_1)}{\sum_{j=1}^{n} \exp(v_j)} & \frac{\exp(v_2)}{\sum_{j=1}^{n} \exp(v_j)} & \cdots & \frac{\exp(v_n)}{\sum_{j=1}^{n} \exp(v_j)} \end{pmatrix}$$

Example: $\begin{pmatrix} 1.5 & 0.3 & -3.7 \end{pmatrix}$

Multinomial logistic regression

How can we map a vector onto the probability simplex?

$$\sigma(\mathbf{v}) = \text{softmax}(\mathbf{v}) := \begin{pmatrix} \frac{\exp(v_1)}{\sum_{j=1}^{n} \exp(v_j)} & \frac{\exp(v_2)}{\sum_{j=1}^{n} \exp(v_j)} & \cdots & \frac{\exp(v_n)}{\sum_{j=1}^{n} \exp(v_j)} \end{pmatrix}$$

Example: $\begin{pmatrix} 1.5 & 0.3 & -3.7 \end{pmatrix}$

Alternatively, we can use the so called one-hot-vector representation

$$\arg\max \begin{pmatrix} 1.5 & 0.3 & -3.7 \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 \end{pmatrix}$$

Multinomial logistic regression

How can we map a vector onto the probability simplex?

$$\sigma(\mathbf{v}) = \text{softmax}(\mathbf{v}) := \left(\frac{\exp(x_1)}{\sum_{j=1}^{n} \exp(v_j)} \quad \frac{\exp(v_2)}{\sum_{j=1}^{n} \exp(v_j)} \quad \cdots \quad \frac{\exp(v_n)}{\sum_{j=1}^{n} \exp(v_j)} \right)$$

Example: $(1.5 \quad 0.3 \quad -3.7)$

Multinomial logistic regression

How can we map a vector onto the probability simplex?

$$\sigma(\mathbf{v}) = \text{softmax}(\mathbf{v}) := \begin{pmatrix} \frac{\exp(x_1)}{\sum_{j=1}^{n} \exp(v_j)} & \frac{\exp(v_2)}{\sum_{j=1}^{n} \exp(v_j)} & \cdots & \frac{\exp(v_n)}{\sum_{j=1}^{n} \exp(v_j)} \end{pmatrix}$$

Example: $\begin{pmatrix} 1.5 & 0.3 & -3.7 \end{pmatrix}$

What if we apply the softmax function to this input?

$$\sigma\left(\begin{pmatrix} 1.5 & 0.3 & -3.7 \end{pmatrix}\right) \approx \begin{pmatrix} 0.765 & 0.231 & 0.004 \end{pmatrix}$$

It is a smoother version of the argmax, hence softmax

Multinomial logistic regression

We can use the softmax function as probability density function for our classification problem:

Multinomial logistic regression

We can use the softmax function as probability density function for our classification problem:

Input/output training samples $\{(\mathbf{x}_i, y_i)\}_{i=1}^{s}$ with $y_i \in \{1, 2, \ldots, K\}$

Multinomial logistic regression

We can use the softmax function as probability density function for our classification problem:

Input/output training samples $\{(\mathbf{x}_i, y_i)\}_{i=1}^{s}$ with $y_i \in \{1,2,\ldots,K\}$

Model assumption: $f(\mathbf{x}_i, \mathbf{w}_1, \ldots, \mathbf{w}_K) = \left(\langle \mathbf{x}_i, \mathbf{w}_1 \rangle \quad \langle \mathbf{x}_i, \mathbf{w}_2 \rangle \quad \ldots \quad \langle \mathbf{x}_i, \mathbf{w}_K \rangle\right)$

Multinomial logistic regression

We can use the softmax function as probability density function for our classification problem:

Input/output training samples $\{(\mathbf{x}_i, y_i)\}_{i=1}^{s}$ with $y_i \in \{1,2,\ldots,K\}$

Model assumption: $f(\mathbf{x}_i, \mathbf{w}_1, \ldots, \mathbf{w}_K) = \left(\langle \mathbf{x}_i, \mathbf{w}_1 \rangle \quad \langle \mathbf{x}_i, \mathbf{w}_2 \rangle \quad \ldots \quad \langle \mathbf{x}_i, \mathbf{w}_K \rangle\right)$

The output is a K dimensional vector for each $\mathbf{x}_i \in \mathbb{R}^{d+1}, \quad \mathbf{w}_k \in \mathbb{R}^{d+1}$

Multinomial logistic regression

Multinomial logistic regression

Likelihood for one pair of samples:

$$\rho(y_i = k \mid \mathbf{x}_i, \mathbf{w}_1, \ldots, \mathbf{w}_K) := \sigma(f(\mathbf{x}_i, \mathbf{w}_1, \ldots, \mathbf{w}_K))_k = \frac{\exp(\langle \mathbf{x}_i, \mathbf{w}_k \rangle)}{\sum_{j=1}^{K} \exp(\langle \mathbf{x}_i, \mathbf{w}_j \rangle)}$$

for $k \in \{1, \ldots, K\}$.

Multinomial logistic regression

Likelihood for all samples:

$$\rho(\hat{\mathbf{y}} = \mathbf{y} \mid \mathbf{X}, \mathbf{W}) := \prod_{i=1}^{s} \rho(\hat{y}_i = y_i \mid \mathbf{x}_i, \mathbf{w}_1, \ldots, \mathbf{w}_K)$$

for $\mathbf{y} = (y_1, \ldots, y_s)^\top$, $\mathbf{X} = \begin{pmatrix} \mathbf{x}_1^\top \\ \vdots \\ \mathbf{x}_s^\top \end{pmatrix}$ and $\mathbf{W} = (\mathbf{w}_1 \quad \mathbf{w}_2 \quad \cdots \quad \mathbf{w}_K)$.

Multinomial logistic regression

$$\rho(\hat{\mathbf{y}} = \mathbf{y} \mid \mathbf{X}, \mathbf{W}) := \prod_{i=1}^{s} \rho(\hat{y}_i = y_i \mid \mathbf{x}_i, \mathbf{w}_1, \ldots, \mathbf{w}_K)$$

Multinomial logistic regression

$$\rho(\hat{\mathbf{y}} = \mathbf{y} \mid \mathbf{X}, \mathbf{W}) := \prod_{i=1}^{s} \rho(\hat{y}_i = y_i \mid \mathbf{x}_i, \mathbf{w}_1, \ldots, \mathbf{w}_K)$$

We can simplify this likelihood as follows:

$$\rho(\hat{\mathbf{y}} = \mathbf{y} \mid \mathbf{X}, \mathbf{W}) = \prod_{\{i \mid y_i = 1\}} \rho(\hat{y}_i = 1 \mid \mathbf{x}_i, \mathbf{w}_1, \ldots, \mathbf{w}_K) \cdots \prod_{\{i \mid y_i = K\}} \rho(\hat{y}_i = K \mid \mathbf{x}_i, \mathbf{w}_1, \ldots, \mathbf{w}_K)$$

Multinomial logistic regression

$$\rho(\hat{\mathbf{y}} = \mathbf{y} \mid \mathbf{X}, \mathbf{W}) := \prod_{i=1}^{s} \rho(\hat{y}_i = y_i \mid \mathbf{x}_i, \mathbf{w}_1, \ldots, \mathbf{w}_K)$$

We can simplify this likelihood as follows:

$$\rho(\hat{\mathbf{y}} = \mathbf{y} \mid \mathbf{X}, \mathbf{W}) = \prod_{\{i \mid y_i = 1\}} \rho(\hat{y}_i = 1 \mid \mathbf{x}_i, \mathbf{w}_1, \ldots, \mathbf{w}_K) \cdots \prod_{\{i \mid y_i = K\}} \rho(\hat{y}_i = K \mid \mathbf{x}_i, \mathbf{w}_1, \ldots, \mathbf{w}_K)$$

We can use the indicator $1_{y_i = k} := \begin{cases} 1 & y_i = k \\ 0 & \text{otherwise} \end{cases}$ notation to simplify the expression above

$$\rho(\hat{\mathbf{y}} = \mathbf{y} \mid \mathbf{X}, \mathbf{W}) := \prod_{i=1}^{s} \prod_{k=1}^{K} \rho(\hat{y}_i = k \mid \mathbf{x}_i, \mathbf{w}_1, \ldots, \mathbf{w}_K)^{1_{y_i = k}}$$

Multinomial logistic regression

$$\rho(\hat{\mathbf{y}} = \mathbf{y} \mid \mathbf{X}, \mathbf{W}) := \prod_{i=1}^{s} \prod_{k=1}^{K} \rho(\hat{y}_i = k \mid \mathbf{x}_i, \mathbf{w}_1, \ldots, \mathbf{w}_K)^{1_{y_i = k}}$$

As usual, we estimate the parameters W by minimising the negative log-likelihood:

$$\hat{\mathbf{W}} = \arg \min_{\mathbf{W} \in \mathbb{R}^{d+1 \times K}} - \log \left(\rho(\hat{\mathbf{y}} = \mathbf{y} \mid \mathbf{X}, \mathbf{W}) \right)$$

Multinomial logistic regression

$$\rho(\hat{\mathbf{y}} = \mathbf{y} \mid \mathbf{X}, \mathbf{W}) := \prod_{i=1}^{s} \prod_{k=1}^{K} \rho(\hat{y}_i = k \mid \mathbf{x}_i, \mathbf{w}_1, \ldots, \mathbf{w}_K)^{1_{y_i=k}}$$

As usual, we estimate the parameters W by minimising the negative log-likelihood:

$$\begin{aligned}
\hat{\mathbf{W}} &= \arg \min_{\mathbf{W} \in \mathbb{R}^{d+1 \times K}} - \log \left(\rho(\hat{\mathbf{y}} = \mathbf{y} \mid \mathbf{X}, \mathbf{W})\right) \\
&= \arg \min_{\mathbf{W} \in \mathbb{R}^{d+1 \times K}} - \log \left(\prod_{i=1}^{s} \prod_{j=1}^{K} \rho(\hat{y}_i = k \mid \mathbf{x}_i, \mathbf{w}_1, \ldots, \mathbf{w}_K)^{1_{y_i=k}} \right)
\end{aligned}$$

Multinomial logistic regression

$$\rho(\hat{\mathbf{y}} = \mathbf{y} \mid \mathbf{X}, \mathbf{W}) := \prod_{i=1}^{s} \prod_{k=1}^{K} \rho(\hat{y}_i = k \mid \mathbf{x}_i, \mathbf{w}_1, \ldots, \mathbf{w}_K)^{1_{y_i=k}}$$

As usual, we estimate the parameters W by minimising the negative log-likelihood:

$$\begin{aligned}
\hat{\mathbf{W}} &= \arg \min_{\mathbf{W} \in \mathbb{R}^{d+1 \times K}} -\log\left(\rho(\hat{\mathbf{y}} = \mathbf{y} \mid \mathbf{X}, \mathbf{W})\right) \\
&= \arg \min_{\mathbf{W} \in \mathbb{R}^{d+1 \times K}} -\log\left(\prod_{i=1}^{s} \prod_{j=1}^{K} \rho(\hat{y}_i = k \mid \mathbf{x}_i, \mathbf{w}_1, \ldots, \mathbf{w}_K)^{1_{y_i=k}}\right) \\
&= \arg \min_{\mathbf{W} \in \mathbb{R}^{d+1 \times K}} -\sum_{i=1}^{s} \sum_{k=1}^{K} 1_{y_i=k} \log\left(\rho(\hat{y}_i = k \mid \mathbf{x}_i, \mathbf{w}_1, \ldots, \mathbf{w}_K)\right)
\end{aligned}$$

Multinomial logistic regression

As usual, we estimate the parameters $\mathbf{W}$ by minimising the negative log-likelihood:

$$\hat{\mathbf{W}} = \arg\min_{\mathbf{W}\in\mathbb{R}^{d+1\times K}} -\sum_{i=1}^{s}\sum_{k=1}^{K} 1_{y_i=k} \log\left(\rho(\hat{y}_i = k \mid \mathbf{x}_i, \mathbf{w}_1, \ldots, \mathbf{w}_K)\right)$$

Multinomial logistic regression

As usual, we estimate the parameters $\mathbf{W}$ by minimising the negative log-likelihood:

$$
\begin{aligned}
\hat{\mathbf{W}} &= \arg\min_{\mathbf{W}\in\mathbb{R}^{d+1\times K}} -\sum_{i=1}^{s}\sum_{k=1}^{K} 1_{y_i=k} \log\left(\rho(\hat{y}_i = k \mid \mathbf{x}_i, \mathbf{w}_1, \ldots, \mathbf{w}_K)\right) \\
&= \arg\min_{\mathbf{W}\in\mathbb{R}^{d+1\times K}} -\sum_{i=1}^{s}\sum_{k=1}^{K} 1_{y_i=k} \log\left(\frac{\exp(\langle \mathbf{x}_i, \mathbf{w}_k\rangle)}{\sum_{j=1}^{K}\exp(\langle \mathbf{x}_i, \mathbf{w}_j\rangle)}\right)
\end{aligned}
$$

Multinomial logistic regression

As usual, we estimate the parameters $\mathbf{W}$ by minimising the negative log-likelihood:

$$\begin{aligned}
\hat{\mathbf{W}} &= \arg \min_{\mathbf{W} \in \mathbb{R}^{d+1 \times K}} - \sum_{i=1}^{s} \sum_{k=1}^{K} 1_{y_i = k} \log \left(\rho(\hat{y}_i = k \mid \mathbf{x}_i, \mathbf{w}_1, \ldots, \mathbf{w}_K) \right) \\
&= \arg \min_{\mathbf{W} \in \mathbb{R}^{d+1 \times K}} - \sum_{i=1}^{s} \sum_{k=1}^{K} 1_{y_i = k} \log \left(\frac{\exp(\langle \mathbf{x}_i, \mathbf{w}_k \rangle)}{\sum_{j=1}^{K} \exp(\langle \mathbf{x}_i, \mathbf{w}_j \rangle)} \right) \\
&= \arg \min_{\mathbf{W} \in \mathbb{R}^{d+1 \times K}} \sum_{i=1}^{s} \sum_{k=1}^{K} 1_{y_i = k} \left(\log \left(\sum_{j=1}^{K} \exp(\langle \mathbf{x}_i, \mathbf{w}_j \rangle) \right) - \langle \mathbf{x}_i, \mathbf{w}_k \rangle \right)
\end{aligned}$$

Multinomial logistic regression

As usual, we estimate the parameters $\mathbf{W}$ by minimising the negative log-likelihood:

$$\hat{\mathbf{W}} = \arg\min_{\mathbf{W} \in \mathbb{R}^{d+1 \times K}} \sum_{i=1}^{s} \sum_{k=1}^{K} 1_{y_i=k} \left(\log \left(\sum_{j=1}^{K} \exp(\langle \mathbf{x}_i, \mathbf{w}_j \rangle) \right) - \langle \mathbf{x}_i, \mathbf{w}_k \rangle \right)$$

Multinomial logistic regression

As usual, we estimate the parameters $\mathbf{W}$ by minimising the negative log-likelihood:

$$\hat{\mathbf{W}} = \arg\min_{\mathbf{W}\in\mathbb{R}^{d+1\times K}} \sum_{i=1}^{s}\sum_{k=1}^{K} 1_{y_i=k}\left(\log\left(\sum_{j=1}^{K}\exp(\langle \mathbf{x}_i, \mathbf{w}_j\rangle)\right) - \langle \mathbf{x}_i, \mathbf{w}_k\rangle\right)$$

$$= \arg\min_{\mathbf{W}\in\mathbb{R}^{d+1\times K}} \sum_{i=1}^{s}\sum_{k=1}^{K} 1_{y_i=k}\log\left(\sum_{j=1}^{K}\exp(\langle \mathbf{x}_i, \mathbf{w}_j\rangle)\right) - \sum_{i=1}^{s}\sum_{k=1}^{K} 1_{y_i=k}\langle \mathbf{x}_i, \mathbf{w}_k\rangle$$

Multinomial logistic regression

As usual, we estimate the parameters $\mathbf{W}$ by minimising the negative log-likelihood:

$$\begin{aligned}
\hat{\mathbf{W}} &= \arg\min_{\mathbf{W}\in\mathbb{R}^{d+1\times K}} \sum_{i=1}^{s}\sum_{k=1}^{K} 1_{y_i=k}\left(\log\left(\sum_{j=1}^{K}\exp(\langle\mathbf{x}_i,\mathbf{w}_j\rangle)\right)-\langle\mathbf{x}_i,\mathbf{w}_k\rangle\right)\\
&= \arg\min_{\mathbf{W}\in\mathbb{R}^{d+1\times K}} \sum_{i=1}^{s}\sum_{k=1}^{K} 1_{y_i=k}\log\left(\sum_{j=1}^{K}\exp(\langle\mathbf{x}_i,\mathbf{w}_j\rangle)\right)-\sum_{i=1}^{s}\sum_{k=1}^{K} 1_{y_i=k}\langle\mathbf{x}_i,\mathbf{w}_k\rangle\\
&= \arg\min_{\mathbf{W}\in\mathbb{R}^{d+1\times K}} \sum_{i=1}^{s}\log\left(\sum_{j=1}^{K}\exp(\langle\mathbf{x}_i,\mathbf{w}_j\rangle)\right)-\sum_{i=1}^{s}\sum_{k=1}^{K} 1_{y_i=k}\langle\mathbf{x}_i,\mathbf{w}_k\rangle
\end{aligned}$$

How to solve logistic regression computationally?

The minimisation problems for logistic regression read

How to solve logistic regression computationally?

The minimisation problems for logistic regression read

Binary: $$\hat{\mathbf{w}} = \arg\min_{w} \left\{ \sum_{i=1}^{s} \log\left(1 + \exp(\langle \mathbf{x}_i, \mathbf{w} \rangle)\right) - y_i \langle \mathbf{x}_i, \mathbf{w} \rangle \right\}$$

How to solve logistic regression computationally?

The minimisation problems for logistic regression read

Binary:
$$\hat{\mathbf{w}} = \arg\min_{w} \left\{ \sum_{i=1}^{s} \log\left(1 + \exp(\langle \mathbf{x}_i, \mathbf{w} \rangle)\right) - y_i \langle \mathbf{x}_i, \mathbf{w} \rangle \right\}$$

Multinomial:
$$\hat{\mathbf{W}} = \arg \min_{\mathbf{W} \in \mathbb{R}^{(d+1)\times K}} \sum_{i=1}^{s} \log\left(\sum_{j=1}^{K} \exp(\langle \mathbf{x}_i, \mathbf{w}_j \rangle) \right) - \sum_{i=1}^{s} \sum_{j=1}^{K} 1_{y_i=j} \langle \mathbf{x}_i, \mathbf{w}_j \rangle$$

How to solve logistic regression computationally?

The minimisation problems for logistic regression read

Binary:
$$\hat{\mathbf{w}} = \arg\min_{w} \left\{ \sum_{i=1}^{s} \log\left(1 + \exp(\langle \mathbf{x}_i, \mathbf{w} \rangle)\right) - y_i \langle \mathbf{x}_i, \mathbf{w} \rangle \right\}$$

Multinomial:
$$\hat{\mathbf{W}} = \arg \min_{\mathbf{W} \in \mathbb{R}^{(d+1)\times K}} \sum_{i=1}^{s} \log\left(\sum_{j=1}^{K} \exp(\langle \mathbf{x}_i, \mathbf{w}_j \rangle) \right) - \sum_{i=1}^{s} \sum_{j=1}^{K} 1_{y_i = j} \langle \mathbf{x}_i, \mathbf{w}_j \rangle$$

How do we solve these minimisation problems computationally?

How to solve logistic regression computationally?

How do we solve these minimisation problems computationally?

Possible approach: gradient descent!

$$\mathbf{w}^{k+1} = \mathbf{w}^k - \tau \nabla L(\mathbf{w}^k)$$

for

$$L(\mathbf{w}^k) = \sum_{i=1}^{s} \log\left(1 + \exp(\langle \mathbf{x}_i, \mathbf{w}^k \rangle)\right) - y_i \langle \mathbf{x}_i, \mathbf{w}^k \rangle$$

How to solve logistic regression computationally?

How do we solve these minimisation problems computationally?

Possible approach: gradient descent!

$$\mathbf{W}^{k+1} = \mathbf{W}^k - \tau \nabla L(\mathbf{W}^k)$$

for

$$L(\mathbf{W}^k) = \sum_{i=1}^{s} \log\left(\sum_{j=1}^{K} \exp(\langle \mathbf{x}_i, \mathbf{w}_j^k \rangle)\right) - \sum_{i=1}^{s} \sum_{j=1}^{K} 1_{y_i = j} \langle \mathbf{x}_i, \mathbf{w}_j^k \rangle$$

How to solve logistic regression computationally?

How do we solve these minimisation problems computationally?

Possible approach: gradient descent!

$$\mathbf{W}^{k+1} = \mathbf{W}^k - \tau \nabla L(\mathbf{W}^k)$$

for

$$L(\mathbf{W}^k) = \sum_{i=1}^{s} \log\left(\sum_{j=1}^{K} \exp(\langle \mathbf{x}_i, \mathbf{w}_j^k \rangle)\right) - \sum_{i=1}^{s} \sum_{j=1}^{K} 1_{y_i=j} \langle \mathbf{x}_i, \mathbf{w}_j^k \rangle$$

For this we need to compute the gradients!

Gradients of logistic regression functions

We compute the gradient for the binary logistic regression case

$$L(\mathbf{w}^k) = \sum_{i=1}^{s} \log\left(1 + \exp(\langle \mathbf{x}_i, \mathbf{w}^k \rangle)\right) - y_i \langle \mathbf{x}_i, \mathbf{w}^k \rangle$$

Gradients of logistic regression functions

We compute the gradient for the binary logistic regression case

$$L(\mathbf{w}^k) = \sum_{i=1}^{s} \log\left(1 + \exp(\langle \mathbf{x}_i, \mathbf{w}^k \rangle)\right) - y_i \langle \mathbf{x}_i, \mathbf{w}^k \rangle$$

Lets start with a simpler problem: for $g(z) := \log(1 + \exp(z))$ we observe

$$g'(z) = \frac{\exp(z)}{1 + \exp(z)} = \frac{1}{1 + \exp(-z)} = \sigma(z)$$

Gradients of logistic regression functions

$$L(\mathbf{w}^k) = \sum_{i=1}^{s} \log\left(1 + \exp(\langle \mathbf{x}_i, \mathbf{w}^k \rangle)\right) - y_i \langle \mathbf{x}_i, \mathbf{w}^k \rangle$$

Hence, we compute the following partial derivatives for the binary logistic regression case:

$$\frac{\partial L}{\partial w_l}(\mathbf{w}^k) = \frac{\partial}{\partial w_l} \sum_{i=1}^{s} \log\left(1 + \exp\left(\sum_{j=0}^{d} x_{ij} w_j^k\right)\right) - y_i \sum_{j=0}^{d} x_{ij} w_j^k$$

$$= \sum_{i=1}^{s} x_{li}^{\top} \sigma\left(\sum_{j=0}^{d} x_{ij} w_j^k\right) - y_i \, x_{il}$$

Gradients of logistic regression functions

$$L(\mathbf{w}^k) = \sum_{i=1}^{s} \log\left(1 + \exp(\langle \mathbf{x}_i, \mathbf{w}^k \rangle)\right) - y_i \langle \mathbf{x}_i, \mathbf{w}^k \rangle$$

As a consequence, the gradient $\nabla L(\mathbf{w}^k)$ reads

$$\nabla L(\mathbf{w}^k) = \mathbf{X}^\top \left(\sigma(\mathbf{X}\mathbf{w}^k) - \mathbf{y}\right)$$

for $\mathbf{X} = \begin{pmatrix} \mathbf{x}_1^\top \\ \vdots \\ \mathbf{x}_s^\top \end{pmatrix}$. Here $\sigma(\mathbf{X}\mathbf{w}^k)$ denotes the application of the logistic function to every component of the vector $\mathbf{X}\mathbf{w}^k$.

Conditions of optimality

Hence, we aim to solve

$$\mathbf{w}^{k+1} = \mathbf{w}^k - \tau \nabla \mathbf{X}^\top \left(\sigma(\mathbf{X}\mathbf{w}^k) - \mathbf{y} \right)$$

to find a weight vector $\hat{\mathbf{w}}$ that satisfies

$$\nabla L(\hat{\mathbf{w}}) = 0 \quad \Leftrightarrow \quad \mathbf{X}^\top \left(\sigma(\mathbf{X}\hat{\mathbf{w}}) - \mathbf{y} \right) = 0$$

Conditions of optimality

Hence, we aim to solve

$$\mathbf{w}^{k+1} = \mathbf{w}^k - \tau \nabla \mathbf{X}^\top \left(\sigma(\mathbf{X}\mathbf{w}^k) - \mathbf{y} \right)$$

to find a weight vector $\hat{\mathbf{w}}$ that satisfies

$$\nabla L(\hat{\mathbf{w}}) = 0 \quad \Leftrightarrow \quad \mathbf{X}^\top \left(\sigma(\mathbf{X}\hat{\mathbf{w}}) - \mathbf{y} \right) = 0$$

We have a numerical procedure, but we also want to know:

$$L(\hat{\mathbf{w}}) \leq L(\mathbf{w}) \quad \forall \mathbf{w} \in \mathbb{R}^{d+1} \, ?$$

Conditions of optimality

Hence, we aim to solve

$$\mathbf{w}^{k+1} = \mathbf{w}^k - \tau \nabla \mathbf{X}^\top \left(\sigma(\mathbf{X}\mathbf{w}^k) - \mathbf{y} \right)$$

to find a weight vector $\hat{\mathbf{w}}$ that satisfies

$$\nabla L(\hat{\mathbf{w}}) = 0 \quad \Leftrightarrow \quad \mathbf{X}^\top \left(\sigma(\mathbf{X}\hat{\mathbf{w}}) - \mathbf{y} \right) = 0$$

We have a numerical procedure, but we also want to know:

$$L(\hat{\mathbf{w}}) \leq L(\mathbf{w}) \quad \forall \mathbf{w} \in \mathbb{R}^{d+1} \ ?$$

How do we find out?

Conditions of optimality

If we can show convexity of L, we already know

$$\nabla L(\hat{\mathbf{w}}) = 0 \quad \Rightarrow \quad L(\hat{\mathbf{w}}) \leq L(\mathbf{w}), \forall \mathbf{w} \in \mathbb{R}^n$$

Conditions of optimality

If we can show convexity of L, we already know

$$\nabla L(\hat{\mathbf{w}}) = 0 \quad \Rightarrow \quad L(\hat{\mathbf{w}}) \leq L(\mathbf{w}), \forall \mathbf{w} \in \mathbb{R}^n$$

Lemma: the function

$$L(\mathbf{w}) \; = \sum_{i=1}^{s} \log\left(1 + \exp(\langle \mathbf{x}_i, \mathbf{w} \rangle)\right) - y_i \langle \mathbf{x}, \mathbf{w} \rangle$$

is convex.

Conditions of optimality

Lemma: the function

$$L(\mathbf{w}) = \sum_{i=1}^{s} \log\left(1 + \exp(\langle \mathbf{x}_i, \mathbf{w} \rangle)\right) - y_i \langle \mathbf{x}, \mathbf{w} \rangle$$

is convex.

Proof:

1. Sum of convex functions is convex
2. The functions $-y_i\langle \mathbf{x}_i, \mathbf{w} \rangle$ are linear in $\mathbf{w}$, and therefore convex
3. We therefore only need to show that

$$\log(1 + \exp(\langle \mathbf{x}_i, \mathbf{w} \rangle))$$

is convex

Conditions of optimality

Lemma: the function

$$L(\mathbf{w}) \ = \sum_{i=1}^{s} \log\left(1 + \exp(\langle \mathbf{x}_i, \mathbf{w}\rangle)\right) - y_i \langle \mathbf{x}, \mathbf{w}\rangle$$

is convex.

Proof continued:

Conditions of optimality

Lemma: the function

$$L(\mathbf{w}) = \sum_{i=1}^{s} \log\left(1 + \exp(\langle \mathbf{x}_i, \mathbf{w} \rangle)\right) - y_i \langle \mathbf{x}, \mathbf{w} \rangle$$

is convex.

Proof continued: consider $f(z) = \log(1 + \exp(z))$

Conditions of optimality

Lemma: the function

$$L(\mathbf{w}) \;=\sum_{i=1}^{s}\log\left(1+\exp(\langle \mathbf{x}_i,\mathbf{w}\rangle)\right)-y_i\langle \mathbf{x},\mathbf{w}\rangle$$

is convex.

Proof continued: consider $f(z)=\log(1+\exp(z))$

We compute $f'(z)=\sigma(z)$ and $f''(z)=\sigma(z)(1-\sigma(z))$

Conditions of optimality

Lemma: the function

$$L(\mathbf{w}) = \sum_{i=1}^{s} \log\left(1 + \exp(\langle \mathbf{x}_i, \mathbf{w} \rangle)\right) - y_i \langle \mathbf{x}, \mathbf{w} \rangle$$

is convex.

Proof continued: consider $f(z) = \log(1 + \exp(z))$

We compute $f'(z) = \sigma(z)$ and $f''(z) = \sigma(z)(1 - \sigma(z))$

We immediately observe $f''(z) \geq 0$ for all $z \in \mathbb{R}$; hence, f is convex

Conditions of optimality

Lemma: the function

$$L(\mathbf{w}) = \sum_{i=1}^{s} \log\left(1 + \exp(\langle \mathbf{x}_i, \mathbf{w} \rangle)\right) - y_i \langle \mathbf{x}, \mathbf{w} \rangle$$

is convex.

Proof continued: consider $f(z) = \log(1 + \exp(z))$

We compute $f'(z) = \sigma(z)$ and $f''(z) = \sigma(z)(1 - \sigma(z))$

We immediately observe $f''(z) \geq 0$ for all $z \in \mathbb{R}$; hence, f is convex

f is a composition of a convex and a linear function and therefore convex ∎

Variational regularisation

Instead of minimising the logistic regression cost function, we can also consider regularised reconstructions:

$$
\begin{aligned}
\hat{\mathbf{w}} &= \arg\min_{\mathbf{w}} \{L(\mathbf{w}) + \alpha R(\mathbf{w})\} \\
&= \arg\min_{\mathbf{w}} \left\{ \sum_{i=1}^{s} \log\big(1 + \exp(\langle \mathbf{x}_i, \mathbf{w}\rangle)\big) - y_i \langle \mathbf{x}_i, \mathbf{w}\rangle + \alpha R(\mathbf{w}) \right\}
\end{aligned}
$$

Variational regularisation

Instead of minimising the logistic regression cost function, we can also consider regularised reconstructions:

$$
\begin{aligned}
\hat{\mathbf{w}} &= \arg\min_{\mathbf{w}} \left\{ L(\mathbf{w}) + \alpha R(\mathbf{w}) \right\} \\
&= \arg\min_{\mathbf{w}} \left\{ \sum_{i=1}^{s} \log\left(1 + \exp(\langle \mathbf{x}_i, \mathbf{w} \rangle)\right) - y_i \langle \mathbf{x}_i, \mathbf{w} \rangle + \alpha R(\mathbf{w}) \right\}
\end{aligned}
$$

Example: logistic ridge regression

$$
\hat{\mathbf{w}} = \arg\min_{w} \left\{ \sum_{i=1}^{s} \left[\log\left(1 + \exp(\langle \mathbf{x}_i, \mathbf{w} \rangle)\right) - y_i \langle \mathbf{x}_i, \mathbf{w} \rangle \right] + \frac{\alpha}{2} \|\mathbf{w}\|^2 \right\}
$$

Variational regularisation

When the regularisation term is also differentiable, then gradient descent can still be applied

$$\nabla L(\mathbf{w}) = X^\top \left(\sigma(\mathbf{X}\mathbf{w}) - \mathbf{y}\right) + \alpha \nabla R(\mathbf{w})$$

Variational regularisation

When the regularisation term is also differentiable, then gradient descent can still be applied

$$\nabla L(\mathbf{w}) = X^\top \left(\sigma(\mathbf{X}\mathbf{w}) - \mathbf{y}\right) + \alpha \nabla R(\mathbf{w})$$

Example: logistic ridge regression

$$\nabla L(\mathbf{w}) = \mathbf{X}^\top \left(\sigma(\mathbf{X}\mathbf{w}) - \mathbf{y}\right) + \alpha\, \mathbf{w}$$

Variational regularisation

When the regularisation term is also differentiable, then gradient descent can still be applied

$$\nabla L(\mathbf{w}) = X^\top \left(\sigma(\mathbf{X}\mathbf{w}) - \mathbf{y}\right) + \alpha \nabla R(\mathbf{w})$$

Example: logistic ridge regression

$$\nabla L(\mathbf{w}) = \mathbf{X}^\top \left(\sigma(\mathbf{X}\mathbf{w}) - \mathbf{y}\right) + \alpha\, \mathbf{w}$$

If R is not differentiable, we can eventually use proximal gradient descent:

$$\mathbf{w}^{k+1} = (I + \tau\, \alpha\, \partial R)^{-1} \left(\mathbf{w}^k - \tau\, \mathbf{X}^\top \left(\sigma(\mathbf{X}\mathbf{w}^k) - \mathbf{y} \right) \right)$$

Link with the logit function

Link with the logit function

Input/output training samples $\{(\mathbf{x}_i, y_i)\}_{i=1}^{s}$ with $y_i \in \{0,1\}$

Link with the logit function

Input/output training samples $\{(\mathbf{x}_i, y_i)\}_{i=1}^{s}$ with $y_i \in \{0,1\}$

Model assumption: $f(\mathbf{x}_i, \mathbf{w}) = \langle \mathbf{x}_i, \mathbf{w} \rangle$

Link with the logit function

Input/output training samples $\{(\mathbf{x}_i, y_i)\}_{i=1}^{s}$ with $y_i \in \{0,1\}$

Model assumption: $f(\mathbf{x}_i, \mathbf{w}) = \langle \mathbf{x}_i, \mathbf{w} \rangle$

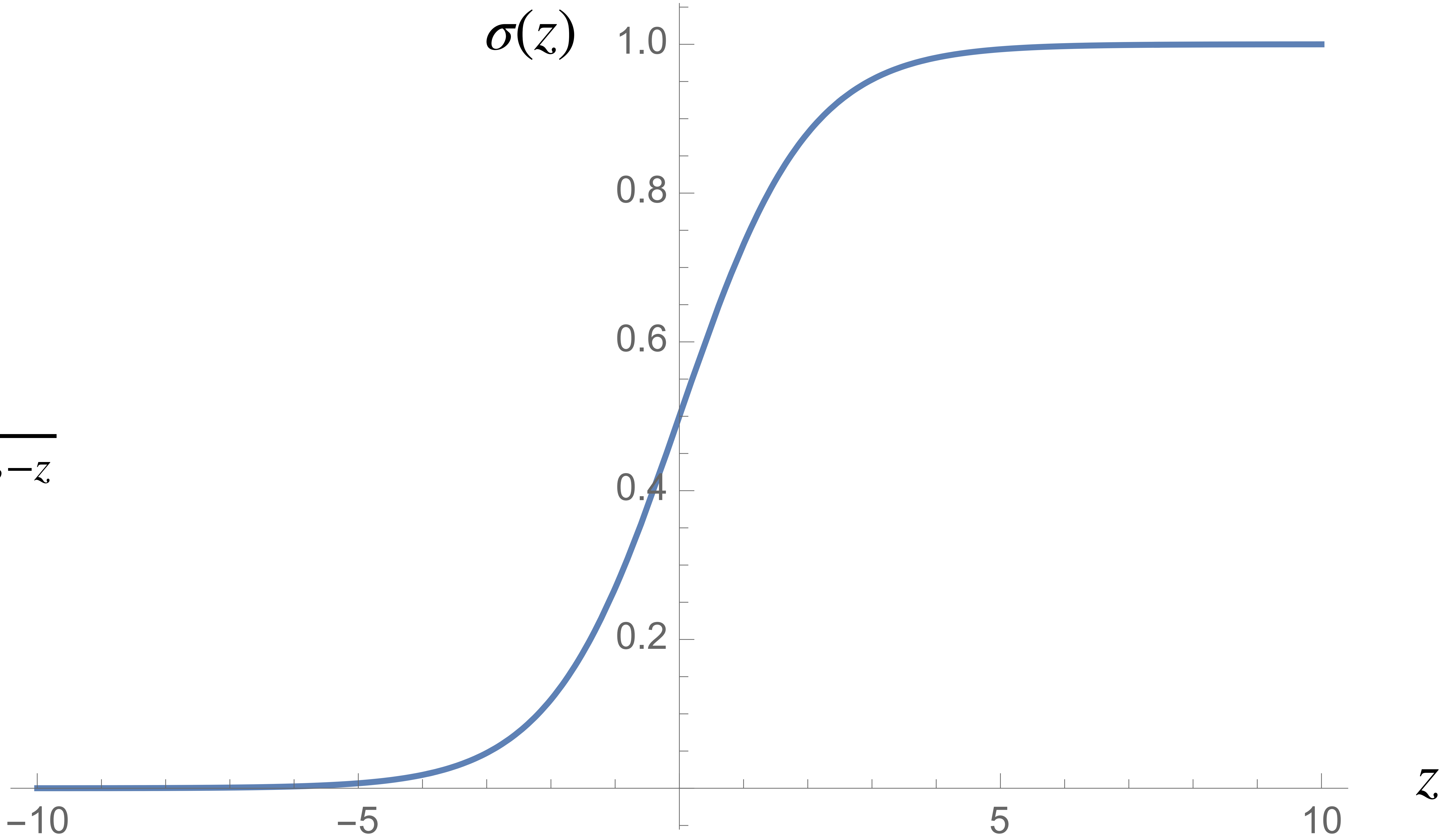
Posterior probability of the two class labels:

$$\rho(1 \,|\, \mathbf{x}_i) = \sigma(\langle \mathbf{x}_i, \mathbf{w} \rangle) \qquad \rho(0 \,|\, \mathbf{x}_i) = 1 - \sigma(\langle \mathbf{x}_i, \mathbf{w} \rangle)$$

for $\quad \mathbf{y} := \begin{pmatrix} y_1 \\ y_2 \\ \vdots \\ y_s \end{pmatrix}, \quad \mathbf{X} := \begin{pmatrix} \mathbf{x}_1^\top \\ \mathbf{x}_2^\top \\ \vdots \\ \mathbf{x}_s^\top \end{pmatrix}$ and $\quad \mathbf{x}_i := \begin{pmatrix} 1 \\ x_{i1} \\ x_{i2} \\ \vdots \\ x_{id} \end{pmatrix}$

Logistic function

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



Link with the logit function

Link with the logit function

Hence, we have $$\rho(1 \,|\, \mathbf{x}_i) = \sigma(\langle \mathbf{x}_i, \mathbf{w} \rangle) = \frac{1}{1 + e^{-\langle \mathbf{x}_i, \mathbf{w} \rangle}}$$

Link with the logit function

Hence, we have $$\rho(1 \mid \mathbf{x}_i) = \sigma(\langle \mathbf{x}_i, \mathbf{w} \rangle) = \frac{1}{1 + e^{-\langle \mathbf{x}_i, \mathbf{w} \rangle}}$$

Can we express the linear model as function of sigma?

Link with the logit function

Hence, we have $$\rho(1 \,|\, \mathbf{x}_i) = \sigma(\langle \mathbf{x}_i, \mathbf{w} \rangle) = \frac{1}{1 + e^{-\langle \mathbf{x}_i, \mathbf{w} \rangle}}$$

Can we express the linear model as function of sigma?

$$1 + e^{-\langle \mathbf{x}_i, \mathbf{w} \rangle} = \frac{1}{\sigma(\langle \mathbf{x}_i, \mathbf{w} \rangle)}$$

Link with the logit function

Hence, we have $$\rho(1\,|\,\mathbf{x}_i) = \sigma(\langle \mathbf{x}_i, \mathbf{w} \rangle) = \frac{1}{1 + e^{-\langle \mathbf{x}_i, \mathbf{w} \rangle}}$$

Can we express the linear model as function of sigma?

$$1 + e^{-\langle \mathbf{x}_i, \mathbf{w} \rangle} = \frac{1}{\sigma(\langle \mathbf{x}_i, \mathbf{w} \rangle)} \quad \rightarrow e^{-\langle \mathbf{x}_i, \mathbf{w} \rangle} = \frac{1}{\sigma(\langle \mathbf{x}_i, \mathbf{w} \rangle)} - 1 = \frac{1 - \sigma(\langle \mathbf{x}_i, \mathbf{w} \rangle)}{\sigma(\langle \mathbf{x}_i, \mathbf{w} \rangle)}$$

Link with the logit function

Hence, we have $$\rho(1 \mid \mathbf{x}_i) = \sigma(\langle \mathbf{x}_i, \mathbf{w} \rangle) = \frac{1}{1 + e^{-\langle \mathbf{x}_i, \mathbf{w} \rangle}}$$

Can we express the linear model as function of sigma?

$$1 + e^{-\langle \mathbf{x}_i, \mathbf{w} \rangle} = \frac{1}{\sigma(\langle \mathbf{x}_i, \mathbf{w} \rangle)} \quad \rightarrow e^{-\langle \mathbf{x}_i, \mathbf{w} \rangle} = \frac{1}{\sigma(\langle \mathbf{x}_i, \mathbf{w} \rangle)} - 1 = \frac{1 - \sigma(\langle \mathbf{x}_i, \mathbf{w} \rangle)}{\sigma(\langle \mathbf{x}_i, \mathbf{w} \rangle)}$$

$$\rightarrow -\langle \mathbf{x}_i, \mathbf{w} \rangle = \ln\left(\frac{1 - \sigma(\langle \mathbf{x}_i, \mathbf{w} \rangle)}{\sigma(\langle \mathbf{x}_i, \mathbf{w} \rangle)}\right)$$

Link with the logit function

Hence, we have $$\rho(1 \,|\, \mathbf{x}_i) = \sigma(\langle \mathbf{x}_i, \mathbf{w} \rangle) = \frac{1}{1 + e^{-\langle \mathbf{x}_i, \mathbf{w} \rangle}}$$

Can we express the linear model as function of sigma?

$$1 + e^{-\langle \mathbf{x}_i, \mathbf{w} \rangle} = \frac{1}{\sigma(\langle \mathbf{x}_i, \mathbf{w} \rangle)} \;\rightarrow e^{-\langle \mathbf{x}_i, \mathbf{w} \rangle} = \frac{1}{\sigma(\langle \mathbf{x}_i, \mathbf{w} \rangle)} - 1 = \frac{1 - \sigma(\langle \mathbf{x}_i, \mathbf{w} \rangle)}{\sigma(\langle \mathbf{x}_i, \mathbf{w} \rangle)}$$

$$\rightarrow -\langle \mathbf{x}_i, \mathbf{w} \rangle = \ln\left(\frac{1 - \sigma(\langle \mathbf{x}_i, \mathbf{w} \rangle)}{\sigma(\langle \mathbf{x}_i, \mathbf{w} \rangle)}\right) \rightarrow \langle \mathbf{x}_i, \mathbf{w} \rangle = \ln\left(\frac{\sigma(\langle \mathbf{x}_i, \mathbf{w} \rangle)}{1 - \sigma(\langle \mathbf{x}_i, \mathbf{w} \rangle)}\right)$$

Link with the logit function

Link with the logit function

We can then write

$$\ln\left(\frac{\sigma(\langle \mathbf{x}_i, \mathbf{w}\rangle)}{1-\sigma(\langle \mathbf{x}_i, \mathbf{w}\rangle)}\right) = \langle \mathbf{x}_i, \mathbf{w}\rangle$$

Link with the logit function

We can then write

$$\ln\left(\frac{\sigma(\langle \mathbf{x}_i, \mathbf{w}\rangle)}{1-\sigma(\langle \mathbf{x}_i, \mathbf{w}\rangle)}\right) = \langle \mathbf{x}_i, \mathbf{w}\rangle$$

Which is equivalent to

$$\ln\left(\frac{\rho(1\,|\,\mathbf{x}_i)}{1-\rho(1\,|\,\mathbf{x}_i)}\right) = logit\left(\rho(1\,|\,\mathbf{x}_i)\right) = \langle \mathbf{x}_i, \mathbf{w}\rangle$$

Link with the logit function

We can then write

$$\ln\left(\frac{\sigma(\langle \mathbf{x}_i, \mathbf{w}\rangle)}{1-\sigma(\langle \mathbf{x}_i, \mathbf{w}\rangle)}\right) = \langle \mathbf{x}_i, \mathbf{w}\rangle$$

Which is equivalent to

$$\ln\left(\frac{\rho(1\,|\,\mathbf{x}_i)}{1-\rho(1\,|\,\mathbf{x}_i)}\right) = logit\left(\rho(1\,|\,\mathbf{x}_i)\right) = \langle \mathbf{x}_i, \mathbf{w}\rangle$$

The left hand side is the log odds since the numerator is the probability of outcome 1 divided by the complementary probability (outcome 0)

Odds ratio

Odds ratio

Writing the problem in this way allows us to provide an easy interpretation of the the weights output of the logistic regression

Odds ratio

Writing the problem in this way allows us to provide an easy interpretation of the the weights output of the logistic regression

We can compute the odds ratio for each variable $p \in [1,d]$

Odds ratio

Writing the problem in this way allows us to provide an easy interpretation of the the weights output of the logistic regression

We can compute the odds ratio for each variable $p \in [1,d]$

$$OR = \frac{odds(\mathbf{x}_i)}{odds(\mathbf{x}_k)} = \frac{e^{\langle \mathbf{x}_i, \mathbf{w} \rangle}}{e^{\langle \mathbf{x}_k, \mathbf{w} \rangle}} = e^{\langle \mathbf{x}_i - \mathbf{x}_k, \mathbf{w} \rangle}$$

Odds ratio

Writing the problem in this way allows us to provide an easy interpretation of the the weights output of the logistic regression

We can compute the odds ratio for each variable $p \in [1,d]$

$$OR = \frac{odds(\mathbf{x}_i)}{odds(\mathbf{x}_k)} = \frac{e^{\langle \mathbf{x}_i, \mathbf{w} \rangle}}{e^{\langle \mathbf{x}_k, \mathbf{w} \rangle}} = e^{\langle \mathbf{x}_i - \mathbf{x}_k, \mathbf{w} \rangle}$$

Let's suppose that the two vectors are the same but for the value in one $p \in [1,d]$

Odds ratio

Writing the problem in this way allows us to provide an easy interpretation of the the weights output of the logistic regression

We can compute the odds ratio for each variable $p \quad \in [1,d]$

$$OR = \frac{odds(\mathbf{x}_i)}{odds(\mathbf{x}_k)} = \frac{e^{\langle \mathbf{x}_i, \mathbf{w} \rangle}}{e^{\langle \mathbf{x}_k, \mathbf{w} \rangle}} = e^{\langle \mathbf{x}_i - \mathbf{x}_k, \mathbf{w} \rangle}$$

Let's suppose that the two vectors are the same but for the value in one $p \in [1,d]$

$$\mathbf{x}_i = (1, x_{i1}, \ldots, x_{ip}, \ldots, x_{id})^\top$$

$$\mathbf{x}_k = (1, x_{i1}, \ldots, x_{ip} - 1, \ldots, x_{id})^\top$$

Odds ratio

Odds ratio

$$OR = \frac{odds(\mathbf{x}_i)}{odds(\mathbf{x}_k)} = \frac{e^{\langle \mathbf{x}_i, \mathbf{w} \rangle}}{e^{\langle \mathbf{x}_k, \mathbf{w} \rangle}} = e^{\langle \mathbf{x}_i - \mathbf{x}_k, \mathbf{w} \rangle}$$

Odds ratio

$$OR = \frac{odds(\mathbf{x}_i)}{odds(\mathbf{x}_k)} = \frac{e^{\langle \mathbf{x}_i, \mathbf{w} \rangle}}{e^{\langle \mathbf{x}_k, \mathbf{w} \rangle}} = e^{\langle \mathbf{x}_i - \mathbf{x}_k, \mathbf{w} \rangle}$$

$$\mathbf{x}_i = (1, x_{i1}, \ldots, x_{ip}, \ldots, x_{id})^\top$$

$$\mathbf{x}_k = (1, x_{i1}, \ldots, x_{ip} - 1, \ldots, x_{id})^\top$$

Odds ratio

$$OR = \frac{odds(\mathbf{x}_i)}{odds(\mathbf{x}_k)} = \frac{e^{\langle \mathbf{x}_i, \mathbf{w} \rangle}}{e^{\langle \mathbf{x}_k, \mathbf{w} \rangle}} = e^{\langle \mathbf{x}_i - \mathbf{x}_k, \mathbf{w} \rangle}$$

$$\mathbf{x}_i = (1, x_{i1}, \ldots, x_{ip}, \ldots, x_{id})^\top$$

$$\mathbf{x}_k = (1, x_{i1}, \ldots, x_{ip} - 1, \ldots, x_{id})^\top$$

$$\mathbf{x}_i - \mathbf{x}_k = (0, 0, \ldots, 1, \ldots, 0)^\top$$

Component p

Odds ratio

$$OR = \frac{odds(\mathbf{x}_i)}{odds(\mathbf{x}_k)} = \frac{e^{\langle \mathbf{x}_i, \mathbf{w} \rangle}}{e^{\langle \mathbf{x}_k, \mathbf{w} \rangle}} = e^{\langle \mathbf{x}_i - \mathbf{x}_k, \mathbf{w} \rangle}$$

$$\mathbf{x}_i = (1, x_{i1}, \ldots, x_{ip}, \ldots, x_{id})^\top$$

$$\mathbf{x}_k = (1, x_{i1}, \ldots, x_{ip} - 1, \ldots, x_{id})^\top$$

$$\mathbf{x}_i - \mathbf{x}_k = (0, 0, \ldots, 1, \ldots, 0)^\top$$

Component p

$$OR_p = \frac{odds(\mathbf{x}_i)}{odds(\mathbf{x}_k)} = \frac{e^{\langle \mathbf{x}_i, \mathbf{w} \rangle}}{e^{\langle \mathbf{x}_k, \mathbf{w} \rangle}} = e^{w_p}$$

Odds ratio

$$OR = \frac{odds(\mathbf{x}_i)}{odds(\mathbf{x}_k)} = \frac{e^{\langle \mathbf{x}_i, \mathbf{w} \rangle}}{e^{\langle \mathbf{x}_k, \mathbf{w} \rangle}} = e^{\langle \mathbf{x}_i - \mathbf{x}_k, \mathbf{w} \rangle}$$

$$\mathbf{x}_i = (1, x_{i1}, \ldots, x_{ip}, \ldots, x_{id})^\top \qquad \mathbf{x}_k = (1, x_{i1}, \ldots, x_{ip} - 1, \ldots, x_{id})^\top$$

$$\mathbf{x}_i - \mathbf{x}_k = (0, 0, \ldots, 1, \ldots, 0)^\top$$

Component p

$$OR_p = \frac{odds(\mathbf{x}_i)}{odds(\mathbf{x}_k)} = \frac{e^{\langle \mathbf{x}_i, \mathbf{w} \rangle}}{e^{\langle \mathbf{x}_k, \mathbf{w} \rangle}} = e^{w_p}$$

This provides an interpretation for the weights: the odds of y=1 are multiplied by e^{w_p} for every unit increase of the variable p

Multinomial logistic regression

$$\rho(y_i = p \mid \mathbf{x}_i, \mathbf{w}_1, \ldots, \mathbf{w}_K) := \sigma(f(\mathbf{x}_i, \mathbf{w}_1, \ldots, \mathbf{w}_K))_p = \frac{\exp(\langle \mathbf{x}_i, \mathbf{w}_p \rangle)}{\sum_{j=1}^{K} \exp(\langle \mathbf{x}_i, \mathbf{w}_j \rangle)}$$

for $p \in \{1, \ldots, K\}$.

Multinomial logistic regression

$$\rho(y_i = p \mid \mathbf{x}_i, \mathbf{w}_1, \ldots, \mathbf{w}_K) := \sigma(f(\mathbf{x}_i, \mathbf{w}_1, \ldots, \mathbf{w}_K))_p = \frac{\exp(\langle \mathbf{x}_i, \mathbf{w}_p \rangle)}{\sum_{j=1}^{K} \exp(\langle \mathbf{x}_i, \mathbf{w}_j \rangle)}$$

for $p \in \{1, \ldots, K\}$.

Since there are K possible outcomes we cannot speak directly about odds ratio as for the binary case

Multinomial logistic regression

What we have seen so far targets the classification task

Multinomial logistic regression

What we have seen so far targets the classification task

If we want to provide an interpretation of the regression bit, we need to switch to odd ratios and logit functions (i.e., multinomial logit regression) which however requires some extra steps

Multinomial logistic regression

In order to be able to define odds with K categories we need to pick one of them (say j) as baseline/reference

Multinomial logistic regression

In order to be able to define odds with K categories we need to pick one of them (say j) as baseline/reference

Hence, in this multinomial logit model, we deal with K-1 binary regressions where we study the odds of each outcome with respect to the baseline

Multinomial logistic regression

In order to be able to define odds with K categories we need to pick one of them (say j) as baseline/reference

Hence, in this multinomial logit model, we deal with K-1 binary regressions where we study the odds of each outcome with respect to the baseline

The baseline is often selected as the most common category, though it is possible to pick any other

Multinomial logistic regression

Hence, we can write, considering j as baseline,

Multinomial logistic regression

Hence, we can write, considering j as baseline,

$$\ln \frac{\rho(y_i = v \,|\, \mathbf{x}_i)}{\rho(y_i = j \,|\, \mathbf{x}_i)} = \langle \mathbf{x}_i, \mathbf{w}_v \rangle$$

Multinomial logistic regression

Hence, we can write, considering j as baseline,

$$\ln \frac{\rho(y_i = v \,|\, \mathbf{x}_i)}{\rho(y_i = j \,|\, \mathbf{x}_i)} = \langle \mathbf{x}_i, \mathbf{w}_v \rangle$$

Which means

$$\frac{\rho(y_i = v \,|\, \mathbf{x}_i)}{\rho(y_i = j \,|\, \mathbf{x}_i)} = e^{\langle \mathbf{x}_i, \mathbf{w}_v \rangle}$$

Multinomial logistic regression

Hence, we can write, considering j as baseline,

$$\ln \frac{\rho(y_i = v \,|\, \mathbf{x}_i)}{\rho(y_i = j \,|\, \mathbf{x}_i)} = \langle \mathbf{x}_i, \mathbf{w}_v \rangle$$

Which means

$$\frac{\rho(y_i = v \,|\, \mathbf{x}_i)}{\rho(y_i = j \,|\, \mathbf{x}_i)} = e^{\langle \mathbf{x}_i, \mathbf{w}_v \rangle}$$

This is the relative risk: the odds of being in category v relative to the reference group j

Multinomial logistic regression

Since $$\sum_{p=1}^{K} \rho(y_i = p \,|\, \mathbf{x}_i) = 1$$

Multinomial logistic regression

Since
$$\sum_{p=1}^{K} \rho(y_i = p \,|\, \mathbf{x}_i) = 1$$

We can write

Multinomial logistic regression

Since $$\sum_{p=1}^{K} \rho(y_i = p \,|\, \mathbf{x}_i) = 1$$

We can write $$\rho(y_i = j \,|\, \mathbf{x}_i) + \sum_{p \neq j} \rho(y_i = p \,|\, \mathbf{x}_i) = 1$$

Multinomial logistic regression

Since
$$\sum_{p=1}^{K} \rho(y_i = p \mid \mathbf{x}_i) = 1$$

We can write
$$\rho(y_i = j \mid \mathbf{x}_i) + \sum_{p \neq j} \rho(y_i = p \mid \mathbf{x}_i) = 1$$

$$\frac{\rho(y_i = p \mid \mathbf{x}_i)}{\rho(y_i = j \mid \mathbf{x}_i)} = e^{\langle \mathbf{x}_i, \mathbf{w}_p \rangle}$$

Multinomial logistic regression

Since $$\sum_{p=1}^{K} \rho(y_i = p \,|\, \mathbf{x}_i) = 1$$

We can write $$\rho(y_i = j \,|\, \mathbf{x}_i) + \sum_{p \neq j} \rho(y_i = p \,|\, \mathbf{x}_i) = 1$$

Using now $$\frac{\rho(y_i = p \,|\, \mathbf{x}_i)}{\rho(y_i = j \,|\, \mathbf{x}_i)} = e^{\langle \mathbf{x}_i, \mathbf{w}_p \rangle} \quad \rightarrow \rho(y_i = p \,|\, \mathbf{x}_i) = \rho(y_i = j \,|\, \mathbf{x}_i) e^{\langle \mathbf{x}_i, \mathbf{w}_p \rangle}$$

Multinomial logistic regression

Since $$\sum_{p=1}^{K} \rho(y_i = p \,|\, \mathbf{x}_i) = 1$$

We can write $$\rho(y_i = j \,|\, \mathbf{x}_i) + \sum_{p \neq j} \rho(y_i = p \,|\, \mathbf{x}_i) = 1$$

Using now $$\frac{\rho(y_i = p \,|\, \mathbf{x}_i)}{\rho(y_i = j \,|\, \mathbf{x}_i)} = e^{\langle \mathbf{x}_i, \mathbf{w}_p \rangle} \quad \rightarrow \rho(y_i = p \,|\, \mathbf{x}_i) = \rho(y_i = j \,|\, \mathbf{x}_i) e^{\langle \mathbf{x}_i, \mathbf{w}_p \rangle}$$

We get

$$\rho(y_i = j \,|\, \mathbf{x}_i) = \frac{1}{1 + \sum_{p \neq j} e^{\langle \mathbf{x}_i, \mathbf{w}_p \rangle}}$$

Multinomial logistic regression

The relative risk for the variable v with respect to the baseline j was

$$\frac{\rho(y_i = v \mid \mathbf{x}_i)}{\rho(y_i = j \mid \mathbf{x}_i)} = e^{\langle \mathbf{x}_i, \mathbf{w}_v \rangle}$$

Multinomial logistic regression

The relative risk for the variable v with respect to the baseline j was

$$\frac{\rho(y_i = v \mid \mathbf{x}_i)}{\rho(y_i = j \mid \mathbf{x}_i)} = e^{\langle \mathbf{x}_i, \mathbf{w}_v \rangle}$$

But since

$$\rho(y_i = j \mid \mathbf{x}_i) = \frac{1}{1 + \sum_{p \neq j} e^{\langle \mathbf{x}_i, \mathbf{w}_p \rangle}}$$

Multinomial logistic regression

The relative risk for the variable v with respect to the baseline j was

$$\frac{\rho(y_i = v \,|\, \mathbf{x}_i)}{\rho(y_i = j \,|\, \mathbf{x}_i)} = e^{\langle \mathbf{x}_i, \mathbf{w}_v \rangle}$$

But since

$$\rho(y_i = j \,|\, \mathbf{x}_i) = \frac{1}{1 + \sum_{p \neq j} e^{\langle \mathbf{x}_i, \mathbf{w}_p \rangle}}$$

We get

$$\rho(y_i = v \,|\, \mathbf{x}_i) = \frac{e^{\langle \mathbf{x}_i, w_v \rangle}}{1 + \sum_{p \neq j} e^{\langle \mathbf{x}_i, \mathbf{w}_p \rangle}}$$

Multinomial logistic regression

The relative risk for the variable v with respect to the baseline j was

$$\frac{\rho(y_i = v \,|\, \mathbf{x}_i)}{\rho(y_i = j \,|\, \mathbf{x}_i)} = e^{\langle \mathbf{x}_i, \mathbf{w}_v \rangle}$$

But since

$$\rho(y_i = j \,|\, \mathbf{x}_i) = \frac{1}{1 + \sum_{p \neq j} e^{\langle \mathbf{x}_i, \mathbf{w}_p \rangle}}$$

We get

$$\rho(y_i = v \,|\, \mathbf{x}_i) = \frac{e^{\langle \mathbf{x}_i, w_v \rangle}}{1 + \sum_{p \neq j} e^{\langle \mathbf{x}_i, \mathbf{w}_p \rangle}}$$

Which is equivalent to the softmax function where we set $w_j = 0$

Multinomial logistic regression

Multinomial logistic regression

You might find references to the relative risk ratio (RRR) which is defined as

Multinomial logistic regression

You might find references to the relative risk ratio (RRR) which is defined as

$$= e^{\langle \mathbf{x}_i - \mathbf{x}_k, \mathbf{w}_v \rangle}$$

Multinomial logistic regression

You might find references to the relative risk ratio (RRR) which is defined as

$$RRR = \frac{\dfrac{\rho(y_i = v \,|\, \mathbf{x}_i)}{\rho(y_i = j \,|\, \mathbf{x}_i)}}{\dfrac{\rho(y_i = v \,|\, \mathbf{x}_k)}{\rho(y_i = j \,|\, \mathbf{x}_k)}} = e^{\langle \mathbf{x}_i - \mathbf{x}_k, \mathbf{w}_v \rangle}$$

Multinomial logistic regression

You might find references to the relative risk ratio (RRR) which is defined as

$$RRR = \frac{\dfrac{\rho(y_i = v \,|\, \mathbf{x}_i)}{\rho(y_i = j \,|\, \mathbf{x}_i)}}{\dfrac{\rho(y_i = v \,|\, \mathbf{x}_k)}{\rho(y_i = j \,|\, \mathbf{x}_k)}} \qquad = e^{\langle \mathbf{x}_i - \mathbf{x}_k, \mathbf{w}_v \rangle}$$

If $\qquad \mathbf{x}_i = (1, x_{i1}, \ldots, x_{im}, \ldots, x_{id})^\top \qquad \mathbf{x}_k = (1, x_{i1}, \ldots, x_{im} - 1, \ldots, x_{id})^\top$

Multinomial logistic regression

You might find references to the relative risk ratio (RRR) which is defined as

$$RRR = \frac{\dfrac{\rho(y_i = v \mid \mathbf{x}_i)}{\rho(y_i = j \mid \mathbf{x}_i)}}{\dfrac{\rho(y_i = v \mid \mathbf{x}_k)}{\rho(y_i = j \mid \mathbf{x}_k)}} = e^{\langle \mathbf{x}_i - \mathbf{x}_k, \mathbf{w}_v \rangle}$$

If $\mathbf{x}_i = (1, x_{i1}, \ldots, x_{im}, \ldots, x_{id})^\top$ $\quad$ $\mathbf{x}_k = (1, x_{i1}, \ldots, x_{im} - 1, \ldots, x_{id})^\top$

$$\mathbf{x}_i - \mathbf{x}_k = (0, 0, \ldots, 1, \ldots, 0)^\top$$

Component m

Multinomial logistic regression

You might find references to the relative risk ratio (RRR) which is defined as

$$RRR = \frac{\dfrac{\rho(y_i = v \mid \mathbf{x}_i)}{\rho(y_i = j \mid \mathbf{x}_i)}}{\dfrac{\rho(y_i = v \mid \mathbf{x}_k)}{\rho(y_i = j \mid \mathbf{x}_k)}} = e^{\langle \mathbf{x}_i - \mathbf{x}_k, \mathbf{w}_v \rangle}$$

If $\mathbf{x}_i = (1, x_{i1}, \ldots, x_{im}, \ldots, x_{id})^\top$ $\mathbf{x}_k = (1, x_{i1}, \ldots, x_{im} - 1, \ldots, x_{id})^\top$

$$\mathbf{x}_i - \mathbf{x}_k = (0, 0, \ldots, 1, \ldots, 0)^\top$$

Component m

$$RRR = e^{\langle \mathbf{x}_i - \mathbf{x}_k, \mathbf{w}_v \rangle} = e^{\mathbf{w}_{vm}}$$