PTML 2: 18/03/2022

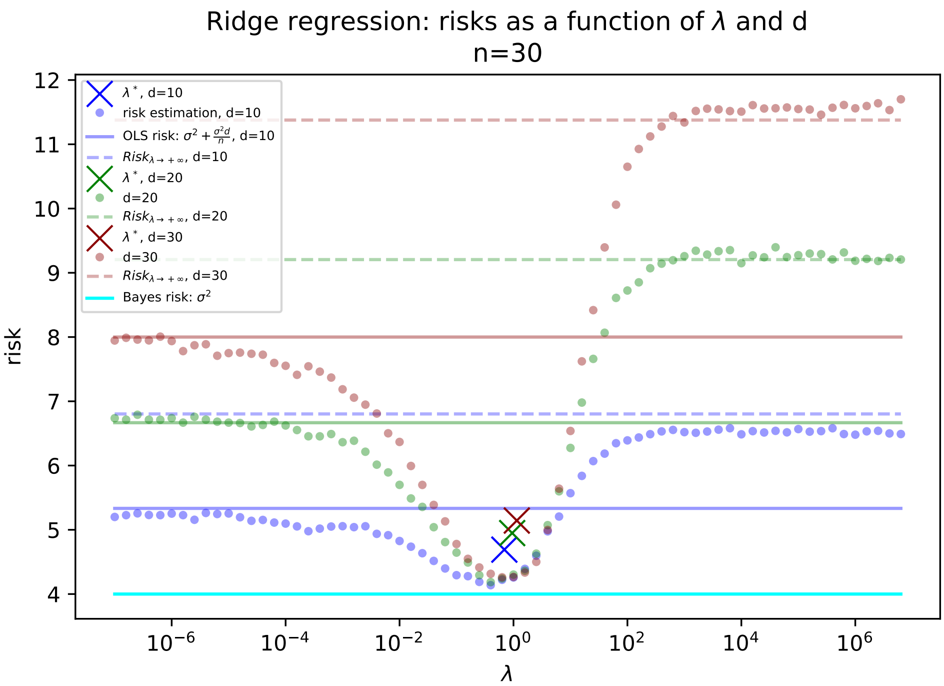

TABLE DES MATIÈRES

1 SOLUTIONS TO EXERCICES 3

See class and FTML/Exercises/Exercices 3 solutions.pdf.

2 COMPARISON OF OLS AND RIDGE REGRESSION

The goal of this exercice is to experimentally observe the benefit of using Ridge regression instead of OLS, in some settings, and to confront observations with the theoretical results.

2.0.1 *Reminders of the theoretical results*

As we have seen in the previous classes, the excess risk in the linear model, fixed design if $\frac{\sigma^2 d}{n}$.

Definition 1. Ridge regression estimator
It is defined as

$$\hat{\theta}_\lambda = \arg\min_{\theta \in \mathbb{R}^d} \left(\frac{1}{n} ||Y - X\theta||_2^2 + \lambda ||\theta||_2^2 \right) \tag{1}$$

Proposition. *The Ridge regression estimator is unique even if* $X^T X$ *is not inversible and is given by*

$$\hat{\theta}_\lambda = \frac{1}{n}(\hat{\Sigma} + \lambda I_d)^{-1} X^T Y$$

Proposition. *Under the linear model assumption, with fixed design setting, the ridge regression estimator has the following excess risk*

$$E[R(\hat{\theta}_\lambda] - R^* = \lambda^2 \theta^{*T} (\hat{\Sigma} + \lambda I_d)^{-2} \hat{\Sigma} \theta^* + \frac{\sigma^2}{n} \mathrm{tr}[\hat{\Sigma}^2 (\hat{\Sigma} + \lambda I_d)^{-2}] \tag{2}$$

Comments :

— We observe again a bias / variance decomposition.

— We consider the bias term B :

$$B = \lambda^2 \theta^{*T} (\hat{\Sigma} + \lambda I_d)^{-2} \hat{\Sigma} \theta^* \tag{3}$$

  — The bias B increases when λ increases. It is an approximation error and does not depend on n.

  — When $\lambda = 0$ and $\hat{\Sigma}$ is invertible (which corresponds to OLS), $B = 0$.

  — When $\lambda \to +\infty$, $B \to \theta^{*T} \hat{\Sigma} \theta^*$.

— We consider the variance term V :

$$V = \frac{\sigma^2}{n} \mathrm{tr}[\hat{\Sigma}^2 (\hat{\Sigma} + \lambda I_d)^{-2}] \tag{4}$$

  — The variance V decreases when λ increases. It is an estimation error and depends on n

  — When $\lambda = 0$ and $\hat{\Sigma}$ is invertible (which corresponds to OLS), $V = \frac{\sigma^2 d}{n}$.

  — When $\lambda \to +\infty$, $V \to 0$.

  — When $n \to +\infty$, $V \to 0$.

— In most cases, it is preferrable to have a biased estimation ($\lambda > 0$).

A natural question is whether it is possible to have a lower excess risk with Ridge regression than with OLS, which means an excess risk smaller than $\frac{\sigma^2 d}{n}$.

Proposition. *With the choice*

$$\lambda^* = \frac{\sigma \sqrt{\mathrm{tr}(\hat{\Sigma})}}{||\theta^*||_2 \sqrt{n}} \tag{5}$$

then

$$E[R(\hat{\theta}_\lambda] - R^* \leqslant \frac{\sigma \sqrt{\mathrm{tr}(\hat{\Sigma})} ||\theta^*||_2}{\sqrt{n}} \tag{6}$$

with

$$\hat{\Sigma} = \frac{1}{n} X^T X \in \mathbb{R}^{d,d} \tag{7}$$

Hence, the convergence to 0 in OLS is in $\frac{1}{n}$, while it is in $\frac{1}{\sqrt{n}}$ for the ridge. However, for the ridge regression, the dependence in the noise if in σ, whereas it is σ^2 for OLS. Which one is preferrable will depend on the value of the constants, and will not necessary be the "fast" rate in $\mathcal{O}(\frac{1}{n})$.

Conclusion : if, for a given setting, we have

$$\frac{\sigma\sqrt{\mathrm{tr}(\hat{\Sigma})}\|\theta^*\|_2}{\sqrt{n}} \leqslant \frac{\sigma^2 d}{n} \tag{8}$$

then we know that there exists values for λ (such as λ^*), such as the Ridge regression estimator has better generalization properties than OLS.

In this exercice we explore such settings.

2.0.2 *First setting*

We assume that $\forall i$, x_i has all its components in $[0, 1]$.

Question 1 : what bound do we have on $\|x_i\|$?

$$\begin{aligned} \|x_i\|^2 &= \sum_{j=1}^{d} ((x_i)_j)^2 \\ &\leqslant \sum_{j=1}^{d} 1 \\ &= d \end{aligned} \tag{9}$$

We deduce that $\forall i$, $\|x_i\| \leqslant d$.

Question 2 : what bound to we have on $\mathrm{tr}(\hat{\Sigma})$?

$$\begin{aligned} \mathrm{tr}(\hat{\Sigma}) &= \sum_{j=1}^{d} \hat{\Sigma}_{jj} \\ &= \frac{1}{n}\sum_{j=1}^{d}\Big(\sum_{i=1}^{n}(x_i)_j^2\Big) \\ &= \frac{1}{n}\sum_{i=1}^{n}\Big(\sum_{j=1}^{d}(x_i)_j^2\Big) \\ &= \frac{1}{n}\sum_{i=1}^{n}\|x_i\|^2 \\ &\leqslant d \end{aligned} \tag{10}$$

We assume that $\theta^* \in [-\frac{1}{\sqrt{d}}, \frac{1}{\sqrt{d}}]^d$.

Question 3 : what bound do we have on $\|\theta^*\|$?

We have by the same calculation as in 9, that $\|\theta^*\| \leqslant 1$

Question 4 : with all these conditions satisfied, how can we ensure that the excess risk of the Ridge estimator for λ^* is smaller than the excess risk of OLS ?

Since

$$\frac{\sigma\sqrt{\operatorname{tr}(\hat{\Sigma})}\|\theta^*\|_2}{\sqrt{n}} \leqslant \frac{\sigma\sqrt{d}}{\sqrt{n}} \tag{11}$$

it would be sufficient to have :

$$\frac{\sigma\sqrt{d}}{\sqrt{n}} \leqslant \frac{\sigma^2 d}{n} \tag{12}$$

which means

$$\frac{\sqrt{n}}{\sigma} \leqslant \sqrt{d} \tag{13}$$

or by squaring

$$\frac{n}{\sigma^2} \leqslant d \tag{14}$$

For instance, if $\sigma = 2$, with $\frac{n}{4} \leqslant d$, the excess risk is smaller for Ridge regression with λ^* than for OLS.

2.0.3 *Simulation 1*

Our goal is to observe the benefit of Ridge compared to OLS in the previous setting, depending on the dimension d, such as in figure 1.

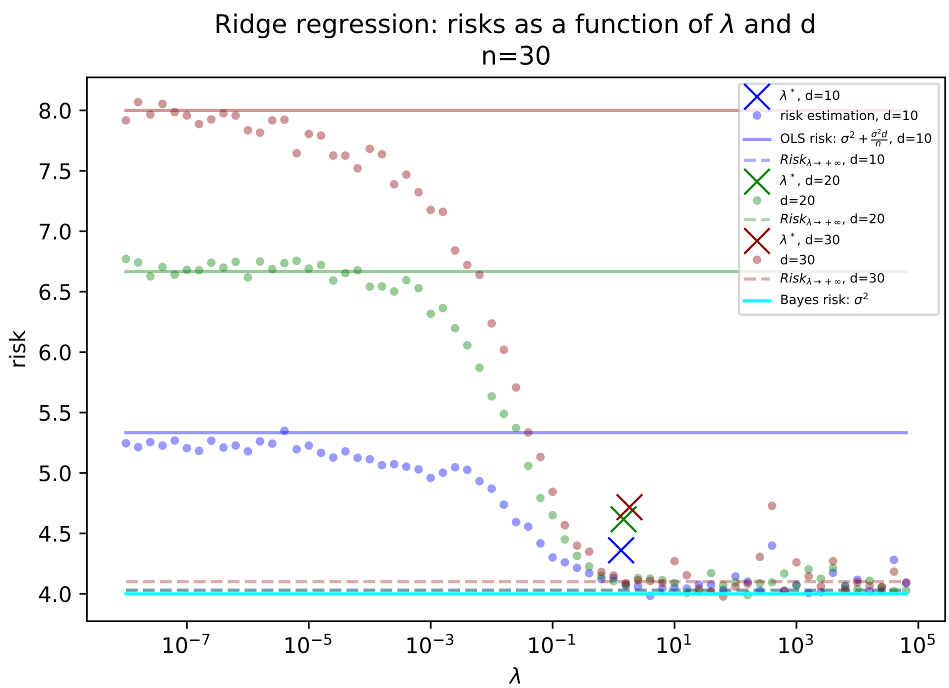

Figure 1 – OLS and ridge

The file to use is **TP_2_ridge_regression.py**

The design matrix is generated by **generate_design _matrix.py**. We use $n = 30$. In order to exhibit the benefit of Ridge, this matrix is **ill-conditioned**. Some columns are almost colinear, leading to a potentially high variance for the OLS estimator, as Σ might have a very small eigenvalues, and thus Σ^{-1} might have some very high eigenvalues.

Step 1 : Initialize θ^* according to the previous setting. (line 175)

Step 2 : Fix **Ridge_estimator()** in order to correctly compute $\hat{\theta}_\lambda$. (line 49)

Step 3 : Fix **compute_lambda_star_and_risk_star()** in order to correctly compute λ^*, and the corresponding risk. (line 82)
Step 4 : Fix the computation of **infinity _bias**, in order to compute the limit of the bias when $\lambda \to +\infty$. (line 184)

2.0.4 *Simulation 2*

The goal of this part is to exhibit a setting where Ridge performs worse than OLS when λ is too large, as in figure 2. As we have seen, when $\lambda \to +\infty$:

— $V \to 0$ (variance)

— $B \to \theta^{*T} \hat{\Sigma} \theta^*$ (bias)

Hence, the excess risk tends to $\theta^{*T} \hat{\Sigma} \theta^*$.

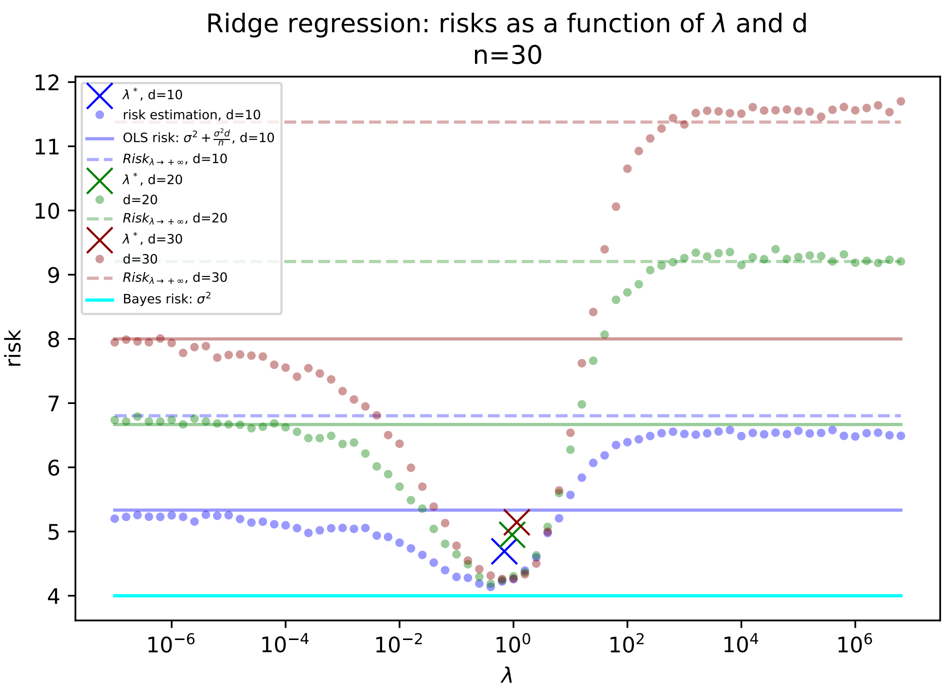

FIGURE 2 – Ridge and OLS, where Ridge performs bad for $\lambda \to +\infty$, because of the bias becomes large.

Question 1 : How could we choose θ^* in order to have a high bias when $\lambda \to +\infty$?

To force a high bias for large λ, an idea would be to force θ^* to be the eigenvector of $\hat{\Sigma}$ with highest eigenvalue.

Step 1 : Initialize θ^* according to this setting. Use the **linalg** module from **numpy**.

3 CROSS VALIDATION

In practical situations, the quantities involved in the computation of λ^* in 5 are typically unknown. Good values for λ are found by **cross-validation.** In fact, there are many variants when applying cross-validation. The theoretical analysis of cross-validation is an active area of research, part of the **model selection** theory.

`https://scikit-learn.org/stable/modules/cross_validation.html` https://en.wikipedia.org/wiki/Cross-validation_(statistics)

We will use **RidgeCV** from scikit-learn. Here, CV stands for cross-validation.
https://scikit-learn.org/stable/modules/generated/sklearn.linear_model.RidgeCV.html#sklearn.linear_model.RidgeCV

Exercise : use scikit-learn and its documentation in order to monitor the values found by cross-validation and compare them to λ^*.

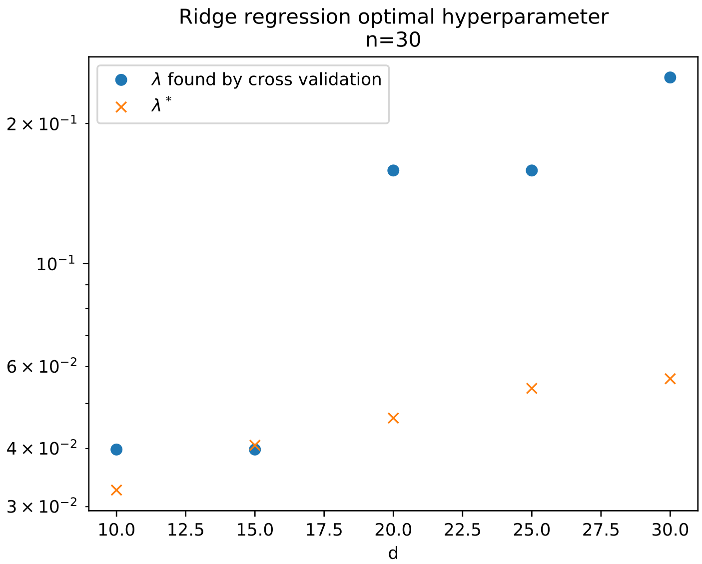

FIGURE 3 – Comparison of λ^* and of the values found by cross-validation, $n = 30$.

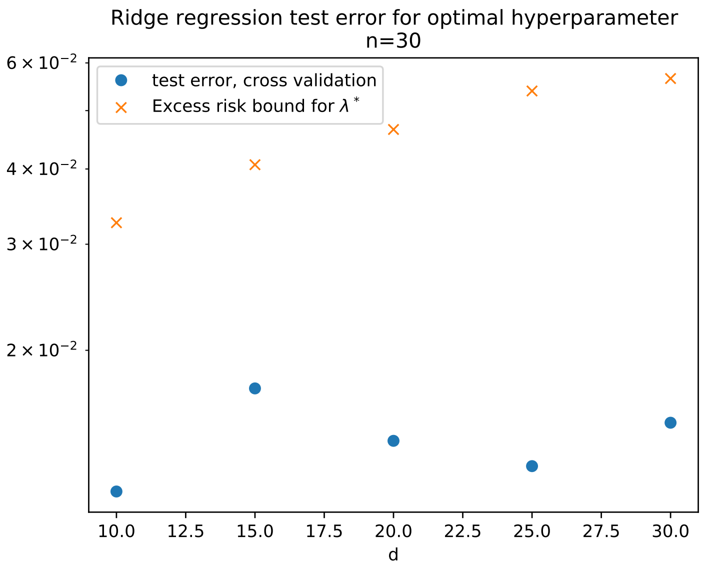

FIGURE 4 – Scores obtained for both parameters, $n = 30$.

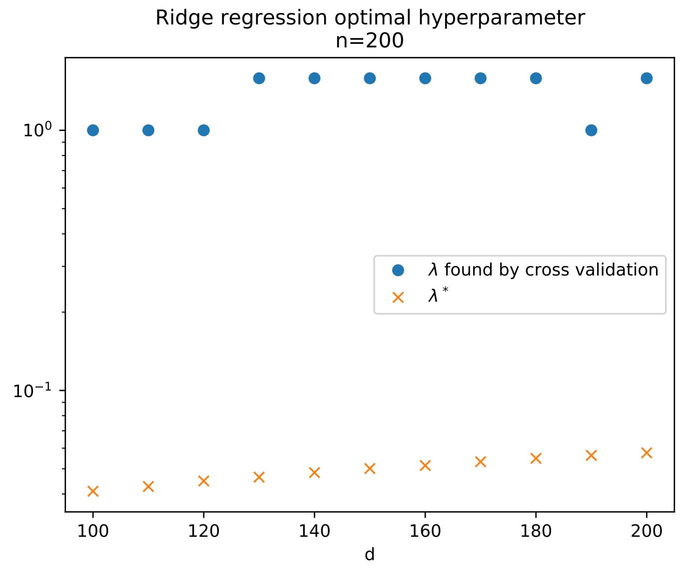

FIGURE 5 – Comparison of λ^* and of the values found by cross-validation, $n = 200$.

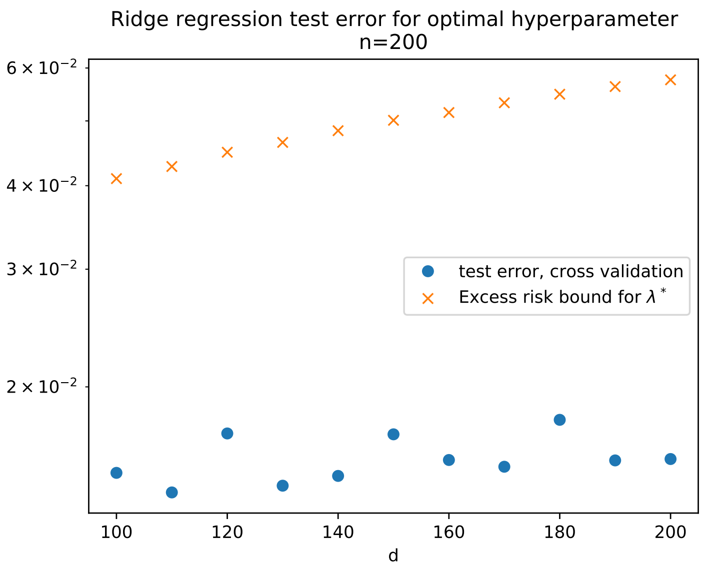

FIGURE 6 – Scores obtained for both parameters, $n = 200$

4 LOGISTIC REGRESSION

Implement a logistic regression estimator optimized with gradient descent, on the dataset **data/gaussian_data.npy**.

Experiment with the parameters of the learning algorithm, and with the dataset. You can modify the dataset with **generate_gaussian_data.py**

https://scikit-learn.org/stable/modules/generated/sklearn.linear_model.LogisticRegression.html#sklearn.linear_model.LogisticRegression

https://scikit-learn.org/stable/modules/generated/sklearn.model_selection.train_test_split.html

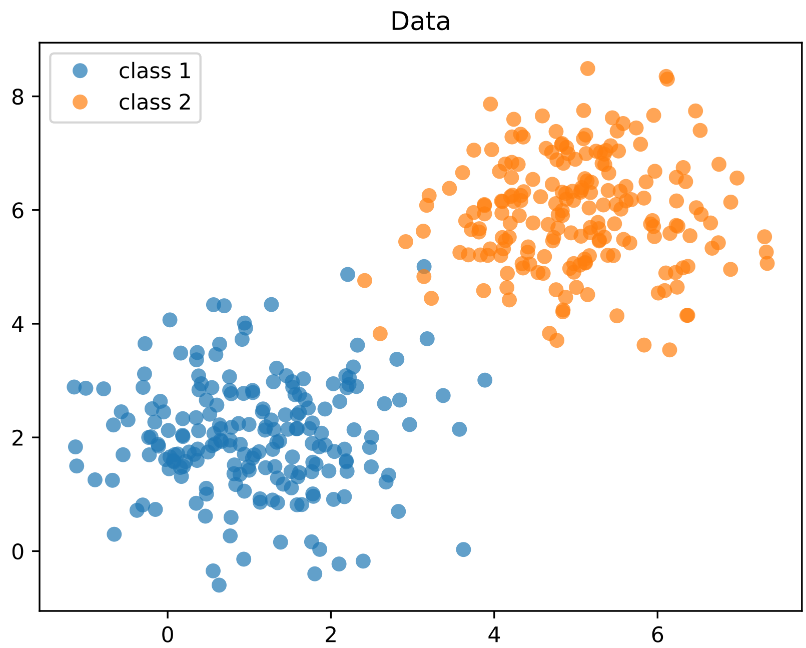

FIGURE 7 – Data to classify

Separate the dataset into a test set and a training set, as in 8

Plot the separator of the obtained decision function, as in figure 9.

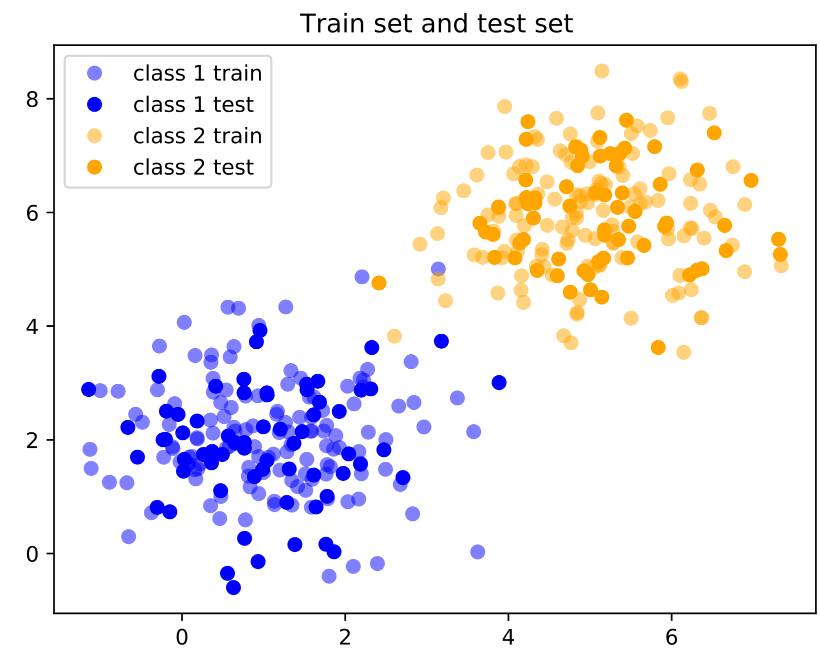

Figure 8 – Test set and train set

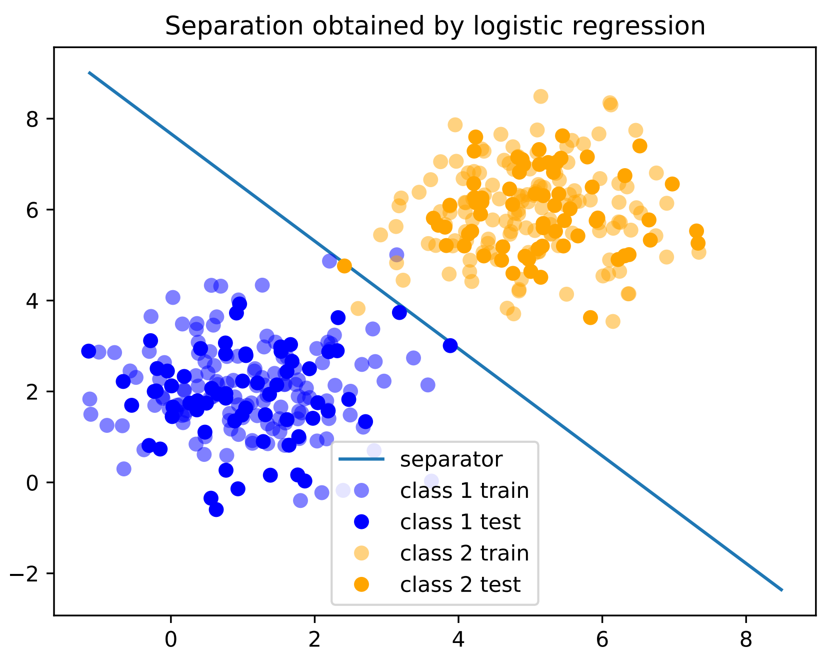

Figure 9 – Decision function