Introduction to machine learning

Project 2

BERNARD Simon (s161519)
KLAPKA Ivan (s165345)

1st year of master in computer science and engineering
Academical year 2019-2020

1 Bayes model and residual error in classification

(a) Analytical formulation of the Bayes model

We start with the definition of the Bayes model

$$\begin{aligned}
h_b(\mathbf{x}) &= \underset{c}{argmax}\ P(y=c|\mathbf{x}) \\
&= \underset{c}{argmax}\ \frac{p(\mathbf{x}|y=c)\,P(y=c)}{p(\mathbf{x})} && \text{using Bayes' theorem} \\
&= \underset{c}{argmax}\ p(\mathbf{x}|y=c)\,P(y=c) && P(\mathbf{x}) \text{ is independent of c} \\
&= \underset{c}{argmax}\ p(\mathbf{x}|y=c) && \text{by the equiprobability of the classes} \\
&= \underset{c}{argmax}\ p(x_0,x_1|y=c)
\end{aligned}$$

Then we make a substitution [1] , replacing (x_0, x_1) by (r, α) with

$$\begin{cases} r = \sqrt{(x_0)^2 + (x_1)^2} \\ \alpha = \text{atan2}(x_1, x_0) \end{cases} \qquad r > 0,\ \alpha \in]-\pi, \pi]$$

Then we get

$$\begin{aligned}
h_b(r,\alpha) &= \underset{c}{argmax}\ p(r,\alpha|y=c) \begin{vmatrix} \cos\alpha & -r\sin\alpha \\ \sin\alpha & r\cos\alpha \end{vmatrix}^{-1} \\
&= \underset{c}{argmax}\ p(r,\alpha|y=c)\,\frac{1}{r} \\
&= \underset{c}{argmax}\ p(r,\alpha|y=c) && \text{r is independant of y} \\
&= \underset{c}{argmax}\ p(r|y=c)\,p(\alpha|y=c) && \text{r and } \alpha \text{ are independant} \\
&= \underset{c}{argmax}\ p(r|y=c)\,p(\alpha) && \alpha \text{ is class-independant} \\
&= \underset{c}{argmax}\ p(r|y=c) && P(\alpha) \text{ is independant of c}
\end{aligned}$$

[1] With the substitution comes the condition $r > 0$, which is not a condition of the distribution of r^i. This means that the following calculations does not consider the fact that a sample could come from a negative r with $\alpha^j = \alpha^i + 180°$. This approximation is acceptable if the overlap of the gaussian distributions are negligible in the negative r. If it is not the case, one should consider $p_{+1}(r) = normalDistr(r, R^+, \sigma) + normalDistr(-r, R^+, \sigma)$ (and with R^- for p_{-1}), but this can not really be simplified and does not lead to a clean condition. Actually, our result are still valid if the negative r probabilities are negligible for $r \leq -\frac{R^+ + R^-}{2}$ which is the limit, that we will found later, between choosing +1 or -1.

We can now write $h_b(\mathbf{x})$ as

$$h_b(\mathbf{x}) = h_b(r) = \begin{cases} +1 & \text{if } \frac{1}{\sigma\sqrt{2\pi}} \exp -\frac{\left(r-R^+\right)^2}{2\sigma^2} > \frac{1}{\sigma\sqrt{2\pi}} \exp -\frac{\left(r-R^-\right)^2}{2\sigma^2} \\ -1 & \text{otherwise} \end{cases}$$

Let's simplify the condition

$$\begin{aligned}
& \frac{1}{\sigma\sqrt{2\pi}} \exp -\frac{(r-R^+)^2}{2\sigma^2} > \frac{1}{\sigma\sqrt{2\pi}} \exp -\frac{(r-R^-)^2}{2\sigma^2} \\
\Leftrightarrow \quad & \exp -\frac{(r-R^+)^2}{2\sigma^2} > \exp -\frac{(r-R^-)^2}{2\sigma^2} \\
\Leftrightarrow \quad & -\frac{(r-R^+)^2}{2\sigma^2} > -\frac{(r-R^-)^2}{2\sigma^2} \\
\Leftrightarrow \quad & \left(r-R^+\right)^2 < \left(r-R^-\right)^2 \\
\Leftrightarrow \quad & \left(r-R^+\right)^2 - \left(r-R^-\right)^2 < 0 \\
\Leftrightarrow \quad & \left[\left(r-R^+\right) + \left(r-R^-\right)\right]\left[\left(r-R^+\right) - \left(r-R^-\right)\right] < 0 \\
\Leftrightarrow \quad & \left(2r-R^+-R^-\right)\left(R^- - R^+\right) < 0
\end{aligned}$$

Assuming R^- is always smaller than R^+ (while the classes distribution only depend on that constant, one can invert them if R^- is bigger than R^+), we get

$$\begin{aligned}
& 2r - R^+ - R^- > 0 \\
\Leftrightarrow \quad & r > \frac{R^+ + R^-}{2}
\end{aligned}$$

Thus we have, replacing r by its (x_0, x_1) value,

$$h_b(\mathbf{x}) = \begin{cases} +1 & \text{if } \sqrt{(x_0)^2 + (x_1)^2} > \frac{R^+ + R^-}{2} \\ -1 & \text{otherwise} \end{cases}$$

(b) Analytical formulation of the residual error

While our Bayes model depend only on r, we can compute the error with respect to the r variable because the error rate will not change if we take α into account.
The error rate with respect to r can be compute as follow

$$\begin{aligned} E_r\left\{\mathbf{1}\left(y \neq y\prime\right)\right\} &= P\left(\hat{y}=-1, y=+1\right)+P\left(\hat{y}=+1, y=-1\right) \\ &= P\left(\hat{y}=-1|y=+1\right)P\left(y=+1\right)+P\left(\hat{y}=+1|y=-1\right)P\left(y=-1\right) \\ &= \left(\int_0^{\frac{R^+ + R^-}{2}} p_{+1}(r)dr\right)\cdot\frac{1}{2}+\left(\int_{\frac{R^+ + R^-}{2}}^{+\infty} p_{-1}(r)dr\right)\cdot\frac{1}{2} \end{aligned}$$

NB^2

With $p_{+1}(r)=\frac{1}{\sigma\sqrt{2\pi}}\exp-\frac{\left(r-R^+\right)^2}{2\sigma^2}$ and $p_{-1}(r)=\frac{1}{\sigma\sqrt{2\pi}}\exp-\frac{\left(r-R^-\right)^2}{2\sigma^2}$

There is no easy solution to those integrals. However, we are able to compute it numerically or we can use the table of a standard normal distribution on $\mathcal{Z}=\frac{\mathcal{N}(R,\ \sigma)-R}{\sigma}$.
So, for $R^-=1$ and $R^+=2$, we have[3]

$$\begin{aligned} E_r\left\{\mathbf{1}\left(y \neq y\prime\right)\right\} &= 0.057\cdot 0.5+0.057\cdot 0.5 \\ &= 5,7\% \end{aligned}$$

2 Bias and variance of the kNN algorithm

(a) Generalization error of kNN

Let's prove that

$$E=E_{LS}\{E_{y|\mathbf{x}}\{(y-\hat{y}(\mathbf{x};LS,k))^2\}\}=\sigma^2+\left[f(\mathbf{x})-\frac{1}{k}\sum_{l=1}^{k}f(\mathbf{x}_{(l)})\right]^2+\frac{\sigma^2}{k} \tag{4}$$

Where $\hat{y}=\hat{y}(\mathbf{x};LS,k)$

First we have :

$$E_{y|\mathbf{x}}\{y\}=f(\mathbf{x})$$

$$\hat{y}=\frac{1}{k}\sum_{l=1}^{k}f(\mathbf{x}_{(l)})+\epsilon_k$$

Thus,

[2]If the negative probabilities are not negligible in $\left]-\frac{R^+ + R^-}{2},0\right]$ but are negligible for $r\leq-\frac{R^+ + R^-}{2}$, we still can have a valid solution taking as a lower bound of the p_{+1} integral $-\frac{R^+ + R^-}{2}$ instead of 0.

[3]`https://homepage.divms.uiowa.edu/~mbognar/applets/normal.html`

$$
\begin{aligned}
E_{LS}\{\hat{y}\} &= \frac{1}{N_{dataset}} \sum_{m=1}^{N_{dataset}} (\frac{1}{k} \sum_{l=1}^{k} f(\mathbf{x}_{(l)}) + \epsilon_k) \\
&= \frac{1}{N_{dataset}} \sum_{m=1}^{N_{dataset}} (\frac{1}{k} \sum_{l=1}^{k} f(\mathbf{x}_{(l)})) + \frac{1}{N_{dataset}} \sum_{m=1}^{N_{dataset}} (\frac{1}{k} \sum_{l=1}^{k} \epsilon_k) \\
&= \frac{1}{N_{dataset}} \sum_{m=1}^{N_{dataset}} (\frac{1}{k} \sum_{l=1}^{k} f(\mathbf{x}_{(l)})) \qquad (5) \\
&= \frac{1}{k} \sum_{l=1}^{k} f(\mathbf{x}_{(l)}) \qquad (6)
\end{aligned}
$$

(5) The expected value of the random variable ϵ over all learning samples is the mean of its normal distribution (assuming the number of learning sets is large enough), which is 0.
(6) As the input values are the same for all learning sets (as assumed in the assignment), the expression $\frac{1}{k} \sum_{l=1}^{k} f(\mathbf{x}_{(l)})$ is constant over all learning sets.

Starting from the initial expression, we now have :

$$
\begin{aligned}
E_{LS}\{E_{y|\mathbf{x}}\{(y-\hat{y})^2\}\} &= E_{LS}\{E_{y|\mathbf{x}}\{(y - E_{y|\mathbf{x}}\{y\} + E_{y|\mathbf{x}}\{y\} - \hat{y})^2\}\} \\
&= E_{LS}\{E_{y|\mathbf{x}}\{(y - E_{y|\mathbf{x}}\{y\})^2\}\} + E_{LS}\{E_{y|\mathbf{x}}\{(E_{y|\mathbf{x}}\{y\} - \hat{y})^2\}\} \\
&+ E_{LS}\{E_{y|\mathbf{x}}\{2(y - E_{y|\mathbf{x}}\{y\})(E_{y|\mathbf{x}}\{y\} - \hat{y})\}\} \\
&= E_{y|\mathbf{x}}\{(y - E_{y|\mathbf{x}}\{y\})^2\} + E_{LS}\{(E_{y|\mathbf{x}}\{y\} - \hat{y})^2\} \qquad (7) \\
&+ E_{LS}\{2(E_{y|\mathbf{x}}\{y\} - E_{y|\mathbf{x}}\{y\})(E_{y|\mathbf{x}}\{y\} - \hat{y})\} \\
&= E_{y|\mathbf{x}}\{(y - E_{y|\mathbf{x}}\{y\})^2\} + E_{LS}\{(E_{y|\mathbf{x}}\{y\} - \hat{y})^2\}
\end{aligned}
$$

(7)

- First term : y is independent of the learning sets
- Second term : $E_{y|\mathbf{x}}\{y\}$ does not depend anymore on $y|\mathbf{x}$ and $\hat{y}$ does not depend on $y|\mathbf{x}$

We can there identify :

$$
\begin{aligned}
\text{The } \textbf{residual error} &= E_{y|\mathbf{x}}\{(y - E_{y|\mathbf{x}}\{y\})^2\} \\
&= var_{y|\mathbf{x}}\{y\} \\
&= var_{y|\mathbf{x}}\{noise\} \\
&= var_{y|\mathbf{x}}\{\epsilon\} \\
&= \sigma^2
\end{aligned}
$$

Let's continue with the second part :

$$
\begin{aligned}
E_{LS}\{(E_{y|\mathbf{x}}\{y\} - \hat{y})^2\} &= E_{LS}\{(E_{y|\mathbf{x}}\{y\} - E_{LS}\{\hat{y}\} + E_{LS}\{\hat{y}\} - \hat{y})^2\} \\
&= E_{LS}\{(E_{y|\mathbf{x}}\{y\} - E_{LS}\{\hat{y}\})^2\} + E_{LS}\{(E_{LS}\{\hat{y}\} - \hat{y})^2\} \\
&+ E_{LS}\{2(E_{y|\mathbf{x}}\{y\} - E_{LS}\{\hat{y}\})(E_{LS}\{\hat{y}\} - \hat{y})\} \\
&= (E_{y|\mathbf{x}}\{y\} - E_{LS}\{\hat{y}\})^2 + E_{LS}\{(\hat{y} - E_{LS}\{\hat{y}\})^2\} \qquad (8) \\
&+ 2(E_{y|\mathbf{x}}\{y\} - E_{LS}\{\hat{y}\})(E_{LS}\{\hat{y}\} - E_{LS}\{\hat{y}\}) \\
&= (E_{y|\mathbf{x}}\{y\} - E_{LS}\{\hat{y}\})^2 + E_{LS}\{(\hat{y} - E_{LS}\{\hat{y}\})^2\}
\end{aligned}
$$

(8) First term : y does not depend on the learning sets and $E_{LS}\{\hat{y}\}$ does not depend anymore on them.

Here we identify two other things :

$$
\begin{aligned}
\text{The } \mathbf{bias}^2 &= (E_{y|\mathbf{x}}\{y\} - E_{LS}\{\hat{y}\})^2 \\
&= \left[f(\mathbf{x}) - \frac{1}{k} \sum_{l=1}^{k} f(\mathbf{x}_{(l)}) \right]^2
\end{aligned}
$$

$$
\begin{aligned}
\text{The } \mathbf{estimation\ variance} &= E_{LS}\left\{(\hat{y} - E_{LS}\{\hat{y}\})^2\right\} \\
&= E_{LS}\left\{ \left[\left(\frac{1}{k} \sum_{l=1}^{k} f(\mathbf{x}_{(l)}) + \epsilon_k \right) - \left(\frac{1}{k} \sum_{l=1}^{k} f(\mathbf{x}_{(l)}) \right) \right]^2 \right\} \\
&= E_{LS}\left\{ \left[\left(\frac{1}{k} \sum_{l=1}^{k} \epsilon_k \right) + \left(\frac{1}{k} \sum_{l=1}^{k} f(\mathbf{x}_{(l)}) \right) - \left(\frac{1}{k} \sum_{l=1}^{k} f(\mathbf{x}_{(l)}) \right) \right]^2 \right\} \\
&= E_{LS}\left\{ \left[\frac{1}{k} \sum_{l=1}^{k} \epsilon_k \right]^2 \right\} \\
&= \frac{1}{k^2} \cdot E_{LS}\left\{ \left[\sum_{l=1}^{k} \epsilon_k \right]^2 \right\} \qquad (9) \\
&= \frac{1}{k^2} \cdot E_{LS}\left\{ \left[\mathcal{N}(k \cdot 0,\ k \cdot \sigma^2) \right]^2 \right\} \qquad (10) \\
&= \frac{1}{k^2} \cdot E_{LS}\left\{ \left[\mathcal{N}(0,\ k\sigma^2) - 0 \right]^2 \right\} \\
&= \frac{1}{k^2} \cdot E_{LS}\left\{ \left[\mathcal{N}(0,\ k\sigma^2) - E_{LS}\left\{ \mathcal{N}(0,\ k\sigma^2) \right\} \right]^2 \right\} \\
&= \frac{1}{k^2} \cdot var_{LS}\left\{ \mathcal{N}(0,\ k\sigma^2) \right\} \qquad (11) \\
&= \frac{1}{k^2} \cdot k\sigma^2 = \frac{\sigma^2}{k}
\end{aligned}
$$

(9) The $E\{\}$ operator is linear.
(10) A random variable which is the sum of independent gaussian random variables is also

a gaussian random variable with a mean equal to the sum of the means and a variance equal to the sum of the variances.
(11) By the definition of the variance.

(b) Effects of the number of neighbours k

The number of neighbours k is present in the **bias**^{2} and in the **estimation variance**.

About the biais^2, the effect is hard to quantify, but we can imagine that taking more and more point that are far from our $\mathbf{x}$ will lead to a worst approximation of the mean (i.e. a worst hypothesis on the model). This would lead to an increase of the error.

About the estimation variance, the more neighbours you take, the closer to the mean of your set you will be and, thus, the less difference you will have within your set. This lead to a decrease of the error.

3 Bias and variance estimation

(a) Protocol to estimate the residual error, the squared bias and the variance

The first step is to compute the residual error for a given $\mathbf{x}_0$. On that purpose, we look inside the learning set and isolate every sample for which $\mathbf{x} = \mathbf{x}_0$. We can then compute the variance off all the different value of y that have the same $\mathbf{x}$: this is the residual error ($E_{y|\mathbf{x}}\{(y - E_{y|\mathbf{x}}\{y\})^2\}$) for a given $\mathbf{x}_0$.

We also compute the mean ($E_{y|\mathbf{x}}\{y\}$) off all y given $\mathbf{x}_0$. This will be used later for calculating the bias.

Next we divide the learning set into n_{LS} smaller sets on which we will train the chosen regression method. We then ask all the different iterations of the method what their prediction $\hat{y}$ for a given $\mathbf{x}_0$ is. We do an average ($E_{LS}\{\hat{y}\}$) of theses prediction over all the different smaller learning sets. With these information we can compute the squared bias ($(E_{y|\mathbf{x}}\{y\} - E_{LS}\{\hat{y}\})^2$)

At last we compute the variance of these predictions ($E_{LS}\left\{(\hat{y} - E_{LS}\{\hat{y}\})^2\right\}$) which correspond to the estimate variance for a given $\mathbf{x}_0$.

For a reason of optimisation, it is not exactly done this way in the code. Indeed, it is easier to handle all different values of $\mathbf{x}_0$ at the same time to avoid training multiple times on the same learning set with the same method. It also allows to browse the learning set only once instead of one time per value of $\mathbf{x}_0$.

(b) Protocol to estimate the mean values of the residual error, the squared bias and the variance

The second protocol simply consists in launching the first protocol for every possible value of $\mathbf{x}$. We then do the average over all of them in order to obtain the mean values of the

residual error ($E_{\mathbf{x}}\{E_{y|\mathbf{x}}\{(y - E_{y|\mathbf{x}}\{y\})^2\}\}$) , the squared bias ($E_{\mathbf{x}}\{(E_{y|\mathbf{x}}\{y\} - E_{LS}\{\hat{y}\})^2\}$) and the variance ($E_{\mathbf{x}}\{E_{LS}\{(\hat{y} - E_{LS}\{\hat{y}\})^2\}\}$).

(c) Discussion for a finite number of samples

The protocols seen above work perfectly with an infinity of samples since every mean or variance have enough samples to show accurate results. In the case of a finite number of samples, things get more complicated.

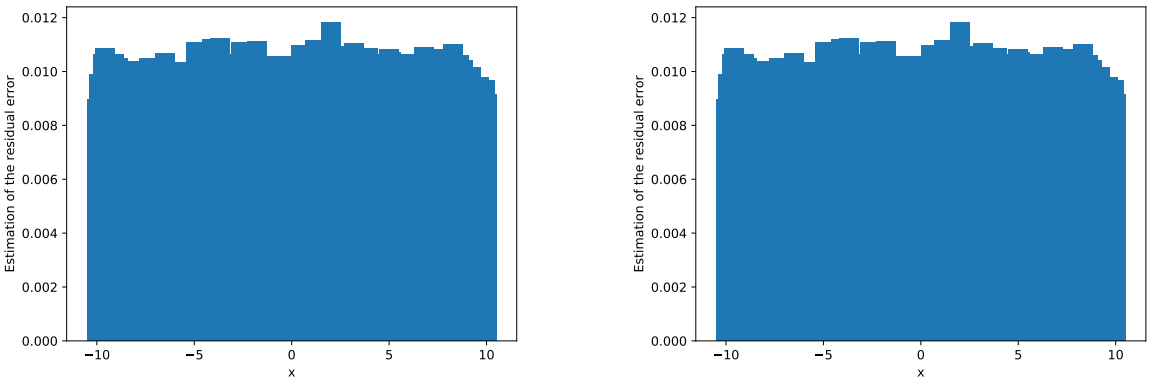
The first problem is that for continuous variable there is an infinity of possible $\mathbf{x}$, this means that we have to approximate and round up or down the values of $\mathbf{x}$ in order to discretise the inputs.

The second problem is that, depending on the number of samples, there might not be enough samples that are close enough to a specific value $\mathbf{x}$. This causes the first protocol to be very inaccurate because every variance and means is done on only one or a few points. In some extreme cases there is no sample for a specific value and thus we are unable to get any information for that value of $\mathbf{x}$.

The only way to limit these issues is to have a larger learning set or to reduce the number of possible states (i.e. discretise on larger intervals or reduce the number of inputs (for the same number of samples)).

(d) Results of the first protocol

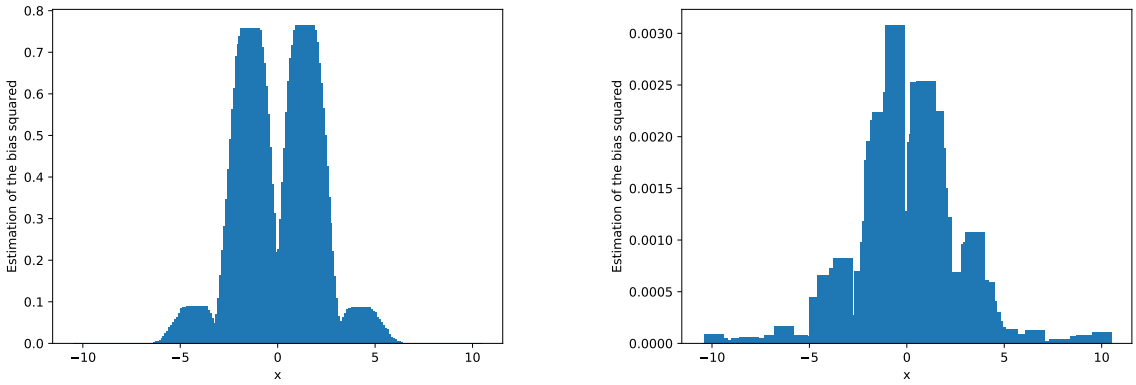
With 100 learning sets of size 1000, we get :



Linear method Non-linear method (kNN, k = 50)

Figure 1 – Residual error with respect to x_r

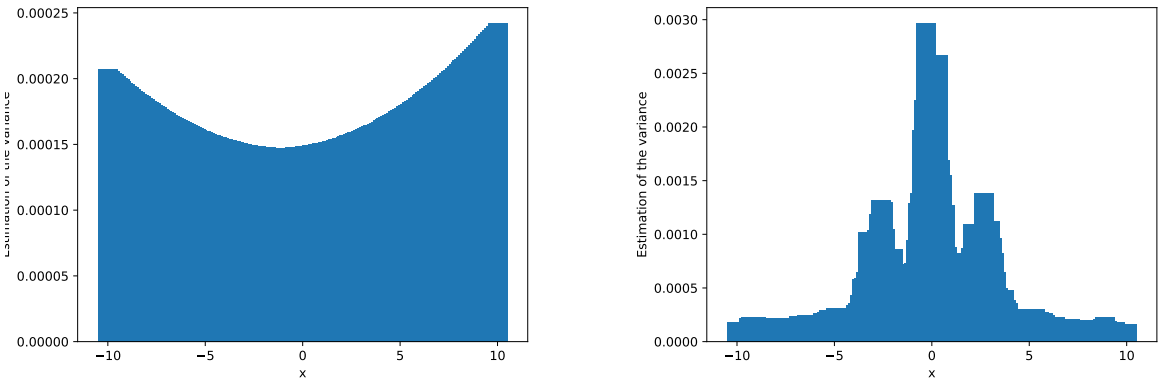
As we can expect, the residual error is more or less constant over the whole space and for both methods. The residual error should not depend on the method as it shows the variance of the observed value itself. Here, we have $var\{\frac{1}{10}\epsilon\} = \frac{1}{100}var\{\epsilon\} = \frac{1}{100} \cdot 1$



Linear method Non-linear method (kNN, k = 50)

Figure 2 – Biais^2 with respect to x_r

As the linear model approximate the exponential decay more or less with a line at $y = 0$ (maybe with a slightly positive slope), the bias^2 is more or less equal to the function itself. This cause the bias^2 to followed the absolute value of the function, getting large error where the function is large.
For the non linear model, as the function observed is non linear, the k nearest neighbours are not equally spread around the true value of y, even though they are around the x observed. This lead to higher bias^2 when closer to x's where the first derivative change fast (e.g. inflection points where the first derivative changes its sign). However, the bias is much smaller than for the linear method.

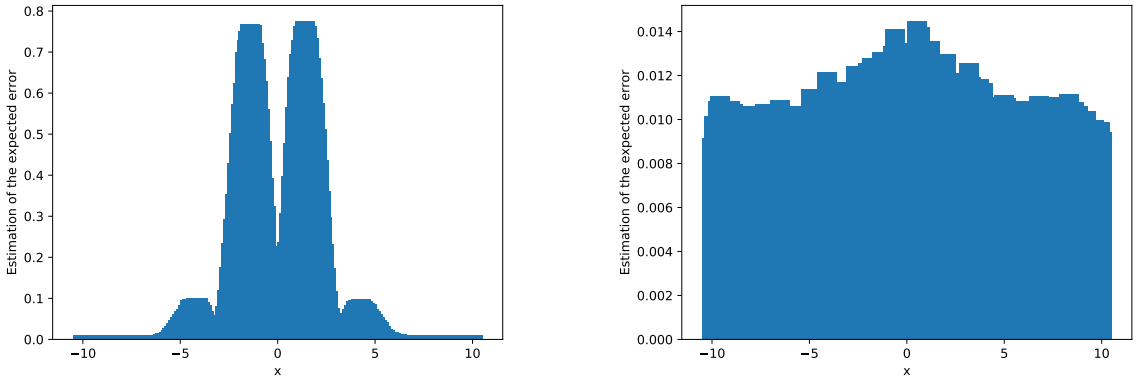


Linear method Non-linear method (kNN, k = 50)

Figure 3 – Estimation variance with respect to x_r

For the linear method, the lines drawed to approximate here will always more or less cross (0, 0) (the more or less explain why the value is not exactly 0 at $x = 0$). It will also have a low positive slope wich explain the slow increase in the limits of the interval (the gap between 2 lines crossing a point increases with respect to the distance from this point and the difference of their slope).

For the non-linear method, the estimation variance is higher where the first derivative of the observed function is high. A slight change in the x value causes a higher difference in its y value, so the mean of the k neareset points has a higher variance.



Linear method Non-linear method (kNN, k = 50)

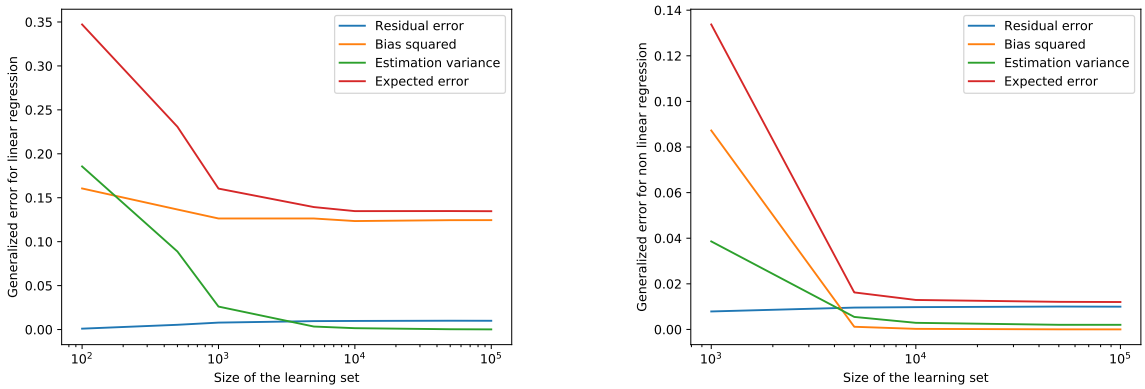
Figure 4 – Expected error with respect to x_r

The first thing to say is well-known, trying to approximate a non linear variable with a linear method often lead to bad result. The "hypothesis" of the linearity is strong and false, such that the bias^2 is huge.
The other fact that we can observe on the non-linear method is that, obviously, the approximation is better for x values around which the output value is more stable.

(e) Results of the second protocol

(e).1 Learning set size

With 100 learning sets, we get :



Linear method Non-linear method (kNN, k = 5)

Figure 5 – Means of the errors with respect to the learning sets size

For the linear method :

Residual error : It is barely equal to 0.01 (what is expected), even for "low" learning set size.

Bias^2 : Quite stable. As it is already huge (due to the bad hypothesis of the linearity), the number of sample does not make a real difference.

Estimation variance : The larger are the learning sets, the more similar they will be. The estimation variance get to zero as the difference between them disappears.

For the non-linear method :

Residual error : Idem as for the linear method (does not depend on the method).

Bias^2 : The more sample you have in the learning set, the closer the k-nearest neighbours will be for each x. Thus, you will be less influenced by far point, which are usually not relevant to estimate your point.

Estimation variance : Idem as for the linear method.

(e).2 Model complexity

With 100 learning sets of size 1000, we get :

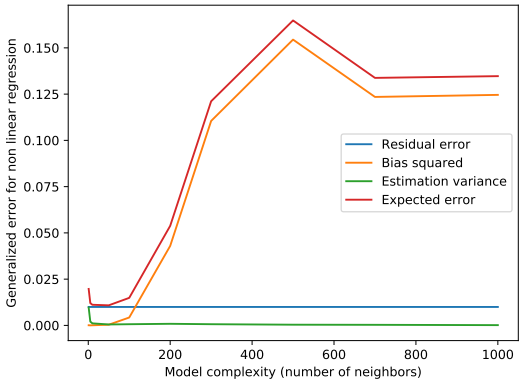


Figure 6 – Non-linear method (kNN)

Figure 7 – Means of the errors with respect to the model complexity

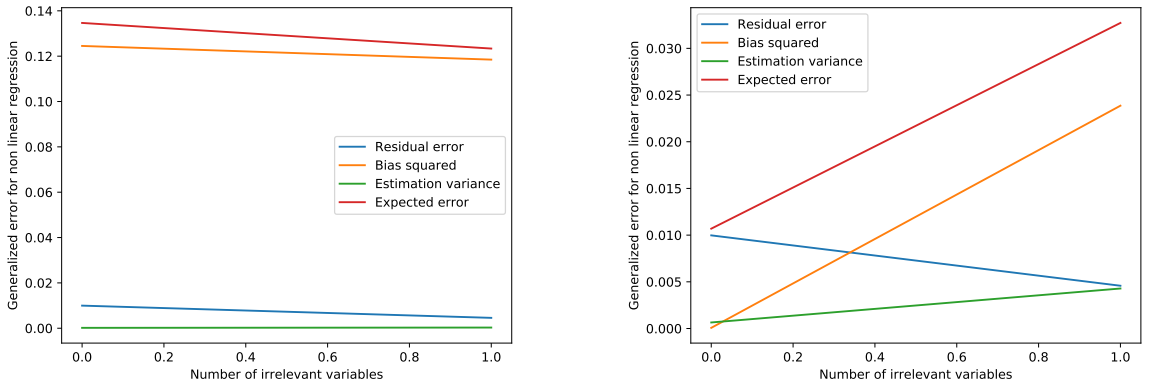
Residual error : It is independant of the model (so also of its complexity).

Bias^2 : It is quite good for a low number of neighbours (i.e. a low complexity) but starts to overfit as we take further neighbours (and stabilize when nearly all points of the sample are considered for every x).

Estimation variance : It is less influenced by the complexity than the bias^2, but we can imagine that the more point you take, the closer to an average you are and the less you have a difference between learning sets.

(e).3 Number of irrelevant features added

With 100 learning sets of size 1000, we get :



Linear method — Non-linear method (kNN, k = 20)

Figure 8 – Means of the errors with respect to the number of irrelevant features added

For the linear method :

Residual error : Because the protocol considers much more "state" as there is more variables, we do not have enough points per state to be able to find the residual error. The lower value for one more feature is artificial and due to the protocol and the learning set size.

Bias^2 : The bias^2 is artificially decreasing (for the same reasons as the residual error), it should increase due to the worst hypothesis of considering the irrelevant features.

Estimation variance : The estimation variance remains more or less stable because having additional inputs does not seems to impact linear methods.

For the non linear method :

Residual error : Same as for the linear method.

Bias^2 : The bias^2 increase due to the worst hypothesis of considering the irrelevant features.

Estimation variance : The estimation variance should increase slightly because point that are far with respect to the relevant feature but close with respect to the irrelevant one could be considered as more interesting because the euclidean distance is lower (e.g. for (x_r, x_1) with x_r the relevant feature, when looking to $(0, 0)$ for example, the point $(20, 0)$ is closer than $(5, 20)$, even though it is much further with respect to the relevant feature). This leads to taking more spread points, i.e. more different values of y, so more variance between learning sets.