ZHAOYANG XU + CHM564

November 26, 2019

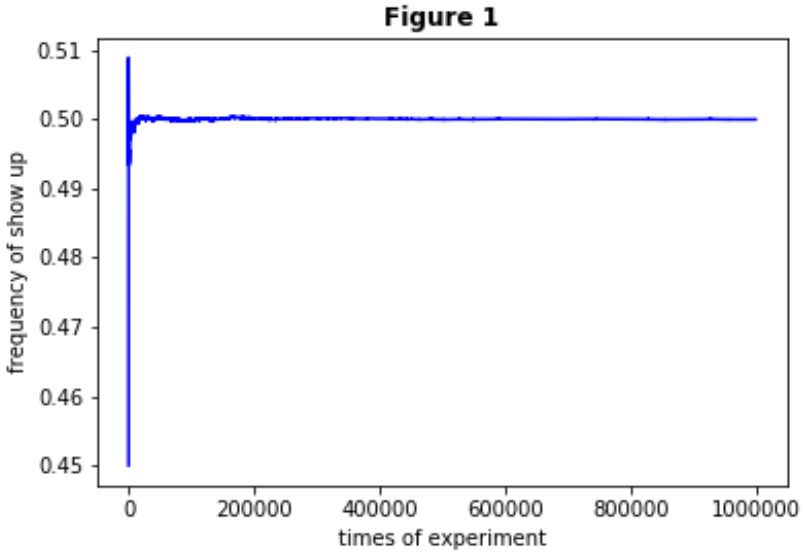
Contents

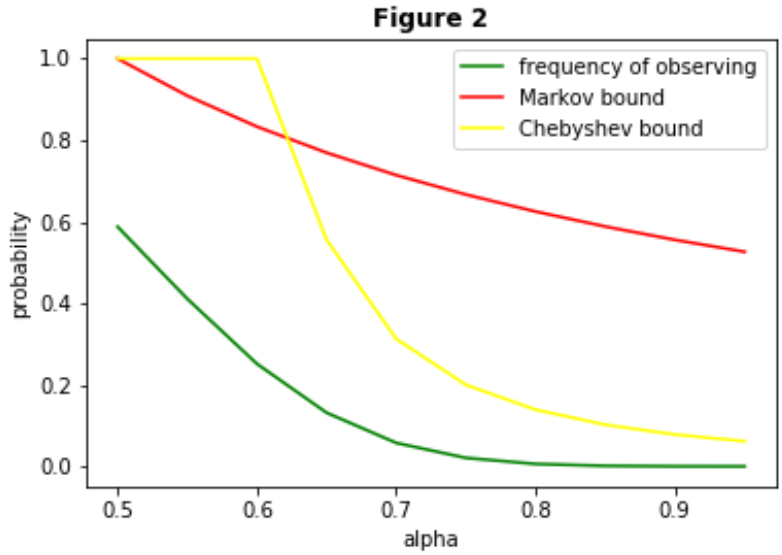
1 Make Your Own

1. We can collect The grades of every home assignments and The grades of some foundation courses(like statistics, computional science) as the sample space $\mathcal{X}$.
2. The grade of final exam as the label space $\mathcal{Y}$.
3. $l(Y', Y) = |Y' - Y|$(absolate loss) We use the difference as the loss.
4. We use the empirical loss $\hat{L}(h, S) = \frac{1}{n}\sum_{i=1}^{n} l(Y_i', Y_i) = \frac{1}{n}\sum_{i=1}^{n} |Y_i' - Y_i|$. Y' is predicting grade ,Y is real grade and n is the number of students. The better the algorithm performed,the smaller the $\hat{L}$ is.
5. Some student may demonstrate progression throughout the course. Or different majors have different difficulties for similar courses. To alleviate them, different assignments and different major can be weighted differently.

2 Illustration of Markov's and Chebychev's Inequalities

3. Because we want to get the empirical frequency of $\frac{1}{20}\sum_{i=1}^{20} X_i \geq \alpha$. So the frequency should be the multiple of 0.05. So take $\alpha = 0.51$ is equal to take $\alpha = 0.55$.
6. From the line of observing(green), we can find that the result always concentrate in the numbers nearby 10.Then we see this line as a reference. We can find that, by Markov's inquality, the probability is much larger than the observing. Meanwhile, with Chebyshev bound, the line decline significantly.And with the α increasing, the difference between the probability in observing and Chebyshev equality become smaller. So, the Chebyshev's equality is much tighter than Markov's.





7. For $\alpha = 1$

$$P(\tfrac{1}{20}\sum_{i=1}^{20} X_i \geq 1) = P(\frac{1}{20}\sum_{i=1}^{20} X_i = 1) = P(\sum_{i=1}^{20} X_i = 20) = P(X_i = 1) = (\frac{1}{2})^{20}$$

For $\alpha = 0.95$

$$P(\tfrac{1}{20}\sum_{i=1}^{20} X_i \geq 0.95) = P(\frac{1}{20}\sum_{i=1}^{20} X_i = 1) + P(\frac{1}{20}\sum_{i=1}^{20} X_i = 0.95)$$
$$= P(\sum_{i=1}^{20} X_i = 20) + P(\sum_{i=1}^{20} X_i = 19) = (\frac{1}{2})^{20} + 20 \times (\frac{1}{2})^{20} = 21 \times (\frac{1}{2})^{21}$$

3 Tightness of Markov's Inequality

Let $X = \{0, 10\}$

	X_1	X_2
X	10	0
P	0	1

$$E[X] = X_1 \times P(X_1) + X_2 \times P(X_2) = 10 \times 0 + 0 \times 1 = 0$$

so $$P(X \geq \varepsilon^*) = \frac{E[X]}{\varepsilon^*} = 0$$

if $\varepsilon^* > 10$ $\quad X = \{10, 0\} < \varepsilon^* \ \Rightarrow \ P(X \geq \varepsilon^*) = 0 = \frac{E[X]}{\varepsilon^*} = 0$

if $10 \geq \varepsilon^* > 0$ $\quad P(X = 10) = 0 = \frac{E[X]}{\varepsilon^*}$

The equality holds a random variable X that accepts $\{0, N\}(N \neq 0)$ with $P(X = 0) = 1;\ P(X = N) = 0$.

4 Digits Classification with Nearest Neighbours

(Because I didn't know the validition error and test error clearly in this question,I use the complete training set and test set to compute test error and first 80% training set and rest 20% training set for validition error. But I also plot the figures with 80%training set with test set and 20%training set for test error and val error.The answer is based on the first situation and I put the rest three figures in the last page)

1. For digits[0,1] : val error match test error well. Before the $K = 31$ the errors always same and equal to 0.

   For digits[0,8] : val error always smaller than test error.Also the two lines even have opposite trend in some K points.

   For digits[5,6] : val error always match well with test error.But not as good as the situation in digits[0,1]. When K choose number nearby 5,7 ,it will be much better.

2. For digits[0,1] : Almost all the K is the best value to val and test error, which is equal to 0.

   For digits[0,8] : For test error the best K value is 1,13,15. At these K. The error is really close between 2 lines. But these are not the best K for val test. Also when val error shows well, test error always large.

For digits[5,6] : The best K for two kinds error are always nearby 3. In $K = 3$, the difference between two error shows close.

3. For digits[0,1] : The two error almost keep 0 for nearly all the K. So we have $\hat{L_{test}} = \hat{L_{val}} = 0 \qquad K = 1, 3, ..., 31$

   For digits[0,8] : Two errors show uneven trend. However,we can see that the val error always much lower. But the lowest error always concentrate nearby 13.

   For digits[5,6] : With the value of K growing, the errors always growing.

4. For digits[0,1], we can choose K for almost almost all the value. However, for[0,8], it's better to chosse some value in the medium. Finally, for[5,6], we can see that it will be better to choose some small value. But when $K = 1$, the error also shows bad. So, it's better to choose some value small, but not smallest. For these data,from the figures, we can see that the task more simple, the range of K we can choose larger. But for other more general situation, I think the best value of K depend on the disturbtion of data.

Figure 3:digitals in [0,1]

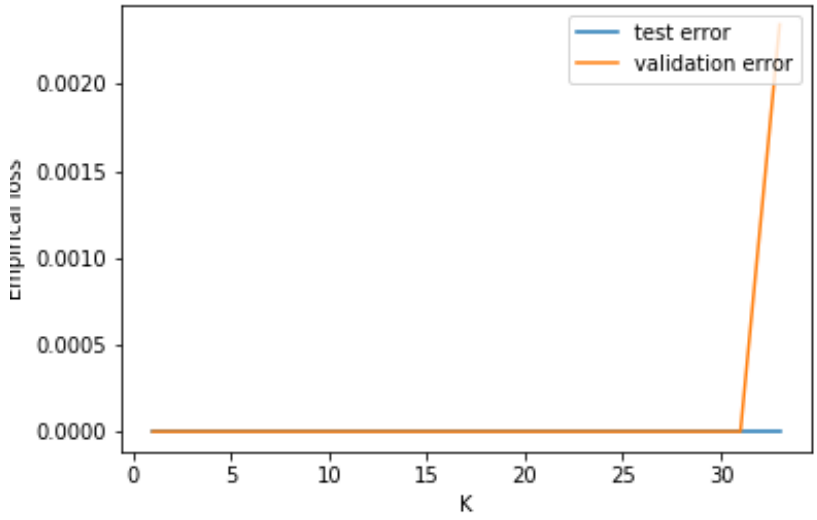


Figure 4:digitals in [0,8]

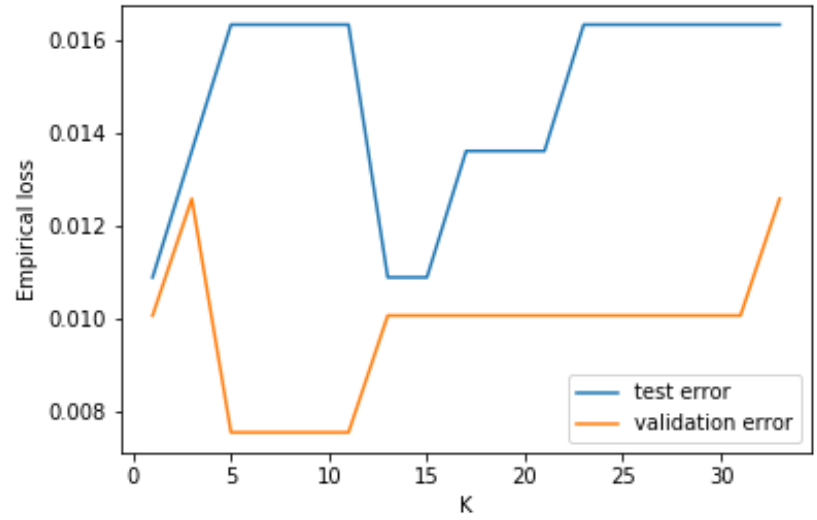
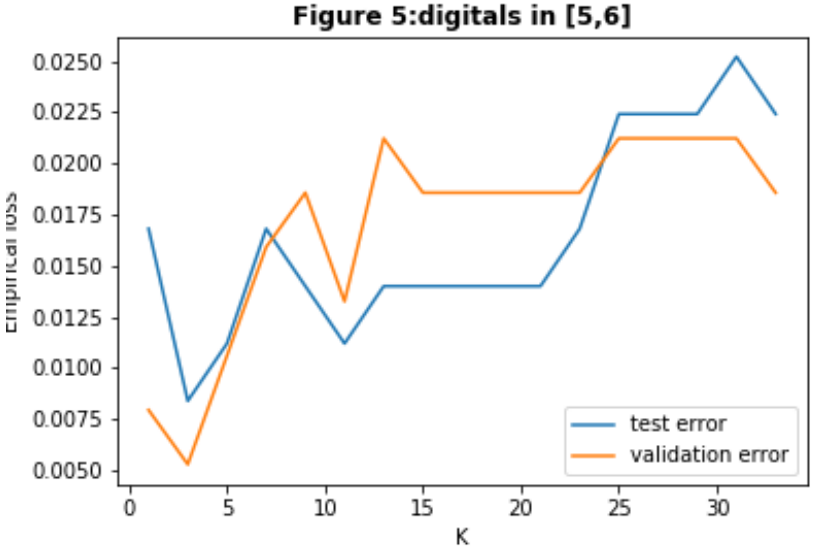


Figure 5:digitals in [5,6]



5 Nearest Neighbours for Multiclass Classification

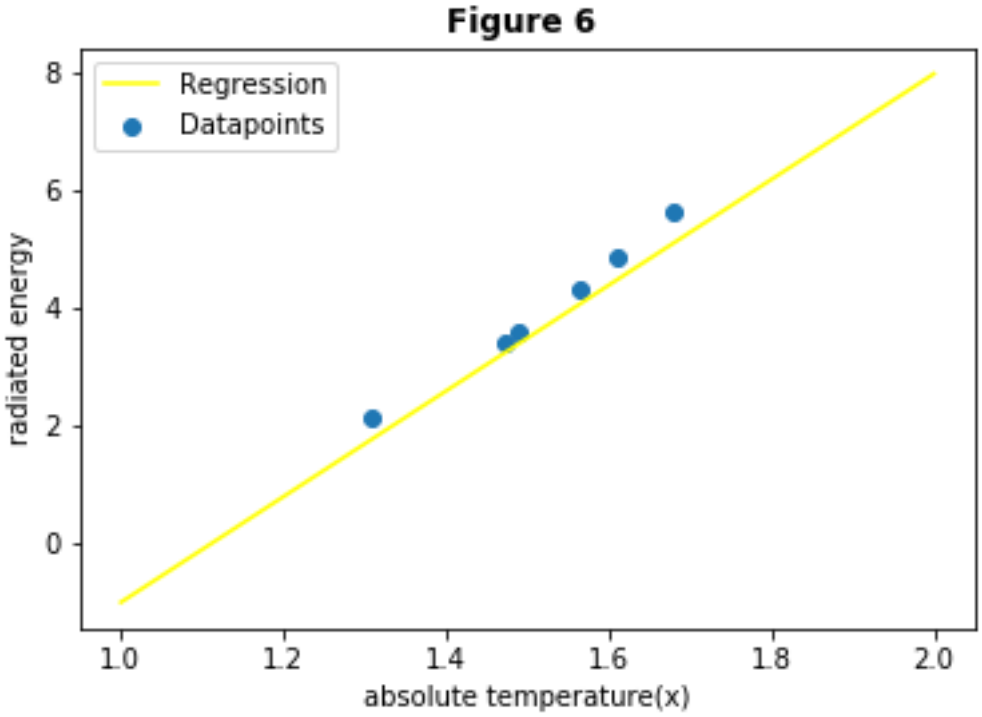
In binary classification, after get K points(which is always odd) from S. Because K is odd, the number of Y_1 must be bigger than the number of Y_2. We can use the majority vote to get Y_1 easily. However, for multiclass classification, we cannot find a good way to choose Y easily, because it is possible to get the same number of Y_i from $\mathcal{Y}$. So, we can modify the way to choose the Y_i. The number of Y_i may be same, but the distance may not. So, we can give different weight to different close points. For example, the number of closest points = 1,the second = $\frac{1}{2}$ and so on.

Algorithm K Nearest Neighbors ($K-NN$) for Multiclass Classification

1:**Input:**A set of labeled points$\{(x_1, y_1), ..., (x_n, y_n)\}$ and a target point x that has to be classified.
2:Calculate the distances $d_i = d(x_i, x)$
3:Sort $d_i - s$ in ascending order and let $\sigma : \{1, ..., n\} \to \{1, ..., n\}$ be the corresponding permutation of indices.
4:Choose the y by $d_i - s$ and orders.

6 Linear Regression

1. See `code.zip`
2. Parameters $w = \begin{pmatrix} 9.48934569 \\ -10.42696146 \end{pmatrix}$ $\quad h : f(x) = 9.48934569x - 10.42696146$
   $MSE = \frac{J}{l} = \frac{1}{l}\sum_{i=1}^{l}\{y_i - f(x_i)\}^2 = 0.01243422161505419$
3. The plot is provided in figure 3.



4. $Var = 1.2689295555555555$
   The log likelihood of the model $h(x) = w_T x$ is
   $lnp(y|w) = ln\prod_{i=1}^{l} \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(y_i - w^T x_i)^2}{2\sigma^2}} = -\frac{l}{2} ln\sigma^2 - \frac{l}{2} ln(2\pi) - \frac{1}{2\sigma^2} \sum_{i=1}^{l} \{y_i - w^T x_i\}^2$

   We know about that, the likelihood probability bigger means the sum-of-squares loss smaller. With first two terms do not depend on w, we just consider the final sum
   $-\frac{1}{2\sigma^2} \sum_{i=1}^{l} \{y_i - w^T x_i\}^2 = -\frac{1}{2} \frac{\frac{1}{l} \sum_{i=1}^{l} \{y_i - w^T x_i\}^2}{\sigma^2} = -\frac{1}{2} \frac{MSE}{Var}$

   if $\frac{MSE}{Var} > 1$ the last item is small,the likelihood probability is small.So the model

   h would not fit well. Confrarily, the $\frac{MSE}{Var} < 1$ the model will fit much better.

5. Parameters $w = \begin{pmatrix} 1.41631595 \\ -1.06635259 \end{pmatrix}$ $z \to x^3$ $h_2 : f(x) = 1.41631595z - 1.06635259$
   $MSE = 0.0004995183326916708$

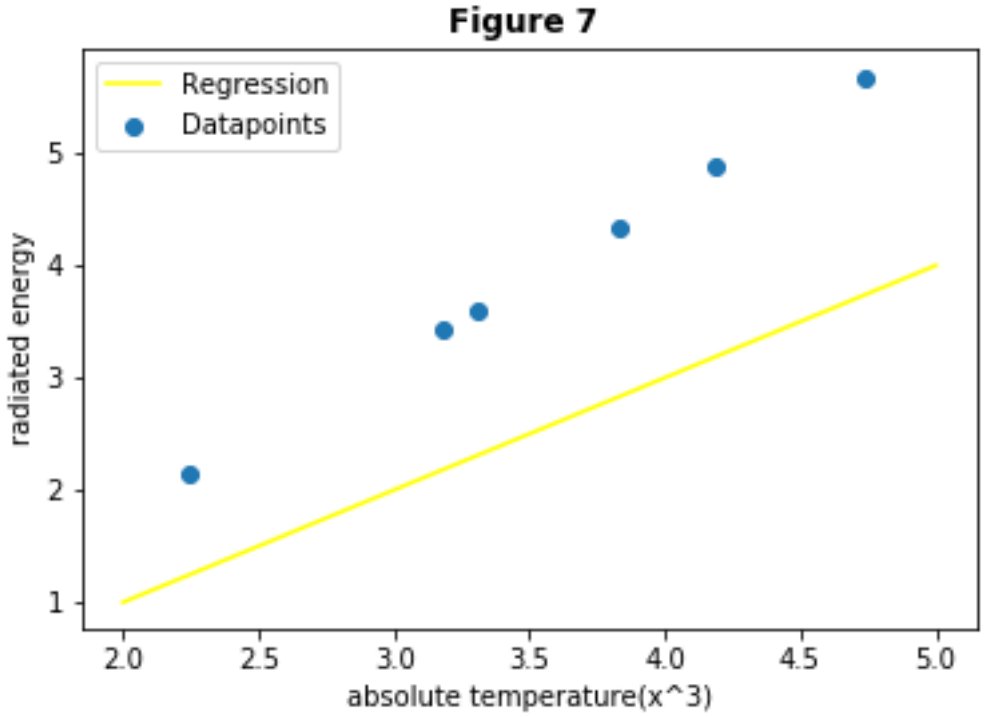


Figure 7
Regression
Datapoints
radiated energy
absolute temperature(x^3)
1
2
3
4
5
2.0
2.5
3.0
3.5
4.0
4.5
5.0

Other figures