CSC2515 Fall 2022: Introduction to Machine Learning Homework 2

Collaborators: Chen Dan and Hongbo Zhou

Student Name: Yulin Wang ID Number: 1003942326

2022-11-1

Q1

(a)

$$\begin{aligned}\frac{1}{n}\sum_{i=1}^{n}|Y_i - m|^2 &= E_D[|Y - m|^2] \\ &= E_D[|Y - E_D[Y] + E_D[Y] - m|^2] \\ &= E_D[|Y - E_D[Y]|^2] + E_D[|E_D[Y] - m|^2] + 2E_D[(Y - E_D[Y])(E_D[Y] - m)]\end{aligned}$$

Since we have

$$E_D[|E_D[Y] - m|^2] = |E_D[Y] - m|^2$$

and

$$\begin{aligned}E_D[(Y - E_D[Y])(E_D[Y] - m)] &= (E_D[Y] - m) \cdot E_D[Y - E_D[Y]] \\ &= (E_D[Y] - m) \cdot (E_D[Y] - E_D[Y]) \\ &= (E_D[Y] - m) \cdot 0 \\ &= 0\end{aligned}$$

Then,

$$\begin{aligned}\frac{1}{n}\sum_{i=1}^{n}|Y_i - m|^2 &= E_D[|Y - E_D[Y]|^2] + |E_D[Y] - m|^2 + 2 \cdot 0 \\ &= Var_D[Y] + |E_D[Y] - m|^2\end{aligned}$$

Thus,

$$\begin{aligned}argmin_{m \in R}\frac{1}{n}\sum_{i=1}^{n}|Y_i - m|^2 &= argmin_{m \in R}\{Var_D[Y] + |E_D[Y] - m|^2\} \\ &= argmin_{m \in R}\{|E_D[Y] - m|^2\}\end{aligned}$$

Therefore, the optimum m is

$$m = E_D[Y] = \frac{1}{n}\sum_{i=1}^{n} Y_i = h_{avg}(D)$$

(b)

The bias of $h_{avg}(D)$:

$$\begin{aligned}
|E_D[h_{avg}(D)] - \mu|^2 &= |E_D[\frac{1}{n}\sum_{i=1}^{n} Y_i - \mu]|^2 \\
&= |\frac{1}{n}\sum_{i=1}^{n} E_D[Y_i] - \mu|^2 \\
&= |\frac{1}{n}\sum_{i=1}^{n} \mu - \mu|^2 \\
&= |\frac{1}{n} \cdot n \cdot \mu - \mu|^2 \\
&= 0
\end{aligned}$$

The variance of $h_{avg}(D)$:

$$\begin{aligned}
Var_D[h_{avg}(D)] &= Var_D[\frac{1}{n}\sum_{i=1}^{n} Y_i] \\
&= \frac{1}{n^2} Var_D[\sum_{i=1}^{n} Y_i] \\
&= \frac{1}{n^2}\sum_{i=1}^{n} Var_D[Y_i] \qquad ; \text{since } Y_i's \text{ are independent} \\
&= \frac{1}{n^2}\sum_{i=1}^{n} \sigma^2 \\
&= \frac{1}{n^2} \cdot n \cdot \sigma^2 \\
&= \frac{\sigma^2}{n}
\end{aligned}$$

(c)

Since from part (a), we have:

$$argmin_{m\in R}\frac{1}{n}\sum_{i=1}^{n}|Y_i - m|^2 = argmin_{m\in R}\{|E_D[Y] - m|^2\}$$

Then,

$$\begin{aligned}
argmin_{m\in R}\frac{1}{n}\sum_{i=1}^{n}|Y_i - m|^2 + \lambda|m|^2 &= argmin_{m\in R}\{|E_D[Y] - m|^2 + \lambda|m|^2\} \\
&= argmin_{m\in R}\{[E_D[Y]]^2 + m^2 - 2mE_D[Y] + \lambda m^2\} \\
&= argmin_{m\in R}\{(1+\lambda)m^2 - 2mE_D[Y]\}
\end{aligned}$$

Let function $J(m) = (1+\lambda)m^2 - 2mE_D[Y]$, and then

$$\begin{aligned}
\frac{dJ(m)}{dm} = 2(1+\lambda)m - 2E_D[Y] &\overset{set}{=} 0 \\
(1+\lambda)m^* &= E_D[Y] \\
h_\lambda(D) = m^* = \frac{E_D[Y]}{1+\lambda} = \frac{h_{avg}(D)}{1+\lambda} &= \frac{1}{n(1+\lambda)}\sum_{i=1}^{n} Y_i
\end{aligned}$$

(d)

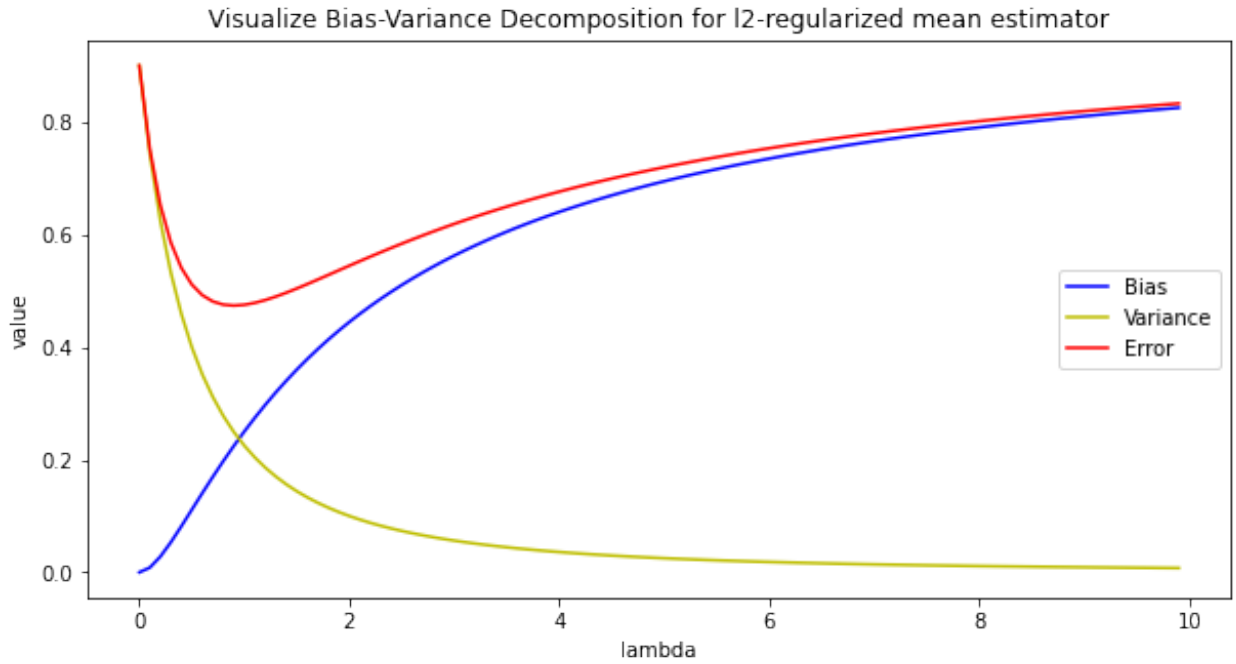
The bias of $h_\lambda(D)$:

$$\begin{aligned}
|E_D[h_\lambda(D)] - \mu|^2 &= |E_D[\frac{1}{n(1+\lambda)}\sum_{i=1}^{n} Y_i - \mu]|^2 \\
&= |\frac{1}{n(1+\lambda)}\sum_{i=1}^{n} E_D[Y_i] - \mu|^2 \\
&= |\frac{1}{n(1+\lambda)}\sum_{i=1}^{n} \mu - \mu|^2 \\
&= |\frac{1}{n(1+\lambda)} \cdot n \cdot \mu - \mu|^2 \\
&= |\frac{\lambda\mu}{(1+\lambda)}|^2
\end{aligned}$$

The variance of $h_\lambda(D)$:

$$\begin{aligned}
Var_D[h_\lambda(D)] &= Var_D[\frac{h_{avg}(D)}{(1+\lambda)}] \\
&= \frac{1}{(1+\lambda)^2} Var_D[h_{avg}(D)] \\
&= \frac{1}{(1+\lambda)^2} \cdot \frac{\sigma^2}{n} \\
&= \frac{\sigma^2}{n(1+\lambda)^2}
\end{aligned}$$

(e)



(f)

Both of the bias and variance contribute to the expected squared error with a trade-off effect. As the lambda increases, the bias increases and the variance decreases, while the error decreases first and then increases, achieving the minimum value when the bias equals to the variance. Moreover, as the lambda increases, the increasing speed of the bias and the decreasing speed of variance become slower, leading to smaller changes on the error.

Q2

(a)

Since $Z_i's$ are identically and independently distributed, so we have the likelihood function:

$$L(\lambda; Z_1, ..., Z_N) = \Pi_{i=1}^N p(Z_i, \lambda) = \Pi_{i=1}^N \frac{\lambda^{Z_i} e^{-\lambda}}{Z_i!}$$

Then the log-likelihood function:

$$\begin{aligned}
l(\lambda; Z_1, ..., Z_N) &= log\{L(\lambda; Z_1, ..., Z_N)\} \\
&= log\{\Pi_{i=1}^N \frac{\lambda^{Z_i} e^{-\lambda}}{Z_i!}\} \\
&= \sum_{i=1}^N log\{\frac{\lambda^{Z_i} e^{-\lambda}}{Z_i!}\} \\
&= \sum_{i=1}^N \{log(\lambda^{Z_i}) + log(e^{-\lambda}) - log(Z_i!)\} \\
&= \sum_{i=1}^N \{Z_i log(\lambda) - \lambda - log(Z_i!)\} \\
&= -N\lambda + log\lambda \sum_{i=1}^N Z_i - \sum_{i=1}^N log(Z_i!)
\end{aligned}$$

Take derivative of the log-likelihood function w.r.t. λ:

$$\begin{aligned}
\frac{dl(\lambda; Z_1, ..., Z_N)}{d\lambda} &= -N + \frac{1}{\lambda} \sum_{i=1}^N Z_i \overset{set}{=} 0 \\
\frac{1}{\lambda^*} \sum_{i=1}^N Z_i &= N \\
\lambda^* &= \frac{1}{N} \sum_{i=1}^N Z_i
\end{aligned}$$

Thus, the maximum likelihood estimate of the λ, $\hat{\lambda}_{MLE} = \frac{1}{N} \sum_{i=1}^N Z_i$, which is the sample mean of $\{Z_1, ..., Z_N\}$.

(b)

Since the data set consisting of i.i.d. input and response variables, so we have the likelihood function:

$$L(w; y_i|x_i) = \Pi_{i=1}^N p(y_i|x_i; w) = \Pi_{i=1}^N \{\frac{exp(y_i w^\top x_i) \cdot exp(-e^{w^\top x_i})}{y_i!}\}$$

Then the log-likelihood function:

$$\begin{aligned} l(w; y_i|x_i) &= log\{L(w; y_i|x_i)\} \\ &= log\{\Pi_{i=1}^{N} \frac{exp(y_i w^\top x_i) \cdot exp(-e^{w^\top x_i})}{y_i!}\} \\ &= \sum_{i=1}^{N} \{log\{exp(y_i w^\top x_i\} + log\{exp(-e^{w^\top x_i})\} - log(y_i!)\} \\ &= \sum_{i=1}^{N} \{y_i w^\top x_i - e^{w^\top x_i} - log(y_i!)\} \end{aligned}$$

Take derivative of the log-likelihood function w.r.t. w:

$$\begin{aligned} \frac{dl(w; y_i|x_i)}{dw} &= \sum_{i=1}^{N} \{y_i x_i - x_i e^{w^\top x_i}\} \\ &= \sum_{i=1}^{N} \{x_i (y_i - e^{w^\top x_i})\} \\ &\overset{set}{=} 0 \end{aligned}$$

Thus, the maximum likelihood resulting estimator $\hat{w}_{MLE}$ satisfies $\sum_{i=1}^{N} \{x_i (y_i - e^{\hat{w}_{MLE}^\top x_i})\} = 0$.

(c)

Write the least squares cost function in matrix form:

$$\begin{aligned} J &= \frac{1}{2}(y - w^\top X)^\top A (y - w^\top X) \\ &= \frac{1}{2}(y^\top - X^\top w) A (y - w^\top X) \\ &= \frac{1}{2}(y^\top A y - X^\top w A y - y^\top A w^\top X + X^\top w A w^\top X) \end{aligned}$$

Differentiate it w.r.t. w:

$$\begin{aligned} \nabla J &= \frac{1}{2}(-X^\top A y - X^\top A y + 2 w X^\top A X) \\ &= -X^\top A y + w X^\top A X \\ &\overset{set}{=} 0 \\ w^* X^\top A X &= X^\top A y \\ w^* &= (X^\top A X)^{-1} X^\top A y \end{aligned}$$

Thus, the solution to this optimization problem is given by the formula $w^* = (X^\top A X)^{-1} X^\top A y$.

(d)

1. As $\tau \to 0$, we only look at the closest data point to query x, because the $a^{(i)}$ of other points will approach to 0, which means that the optimization problem will be solved by getting the label of the closest point to each query x in the training set. This behaves like a K-NN classifier with K = 1.
2. As $\tau \to \infty$, the numerator of $a^{(i)}$ approaches to 1 and the denominator approaches to N, then $a^{(i)} \to \frac{1}{N}$. Thus, the weight for each query x will be approximately the same as $\frac{1}{N}$, which behaves like an ordinary linear regression.
3. Compared to the least squares regression, this approach is more computationally expensive and more complex. Because for every query x, it needs to run the training process on the whole training set to compute the corresponding weights.

Q3.1 Initial data analysis

(a)

Please refer to my uploaded file CSC2515_HW2.ipynb on MarkUs.

(b)

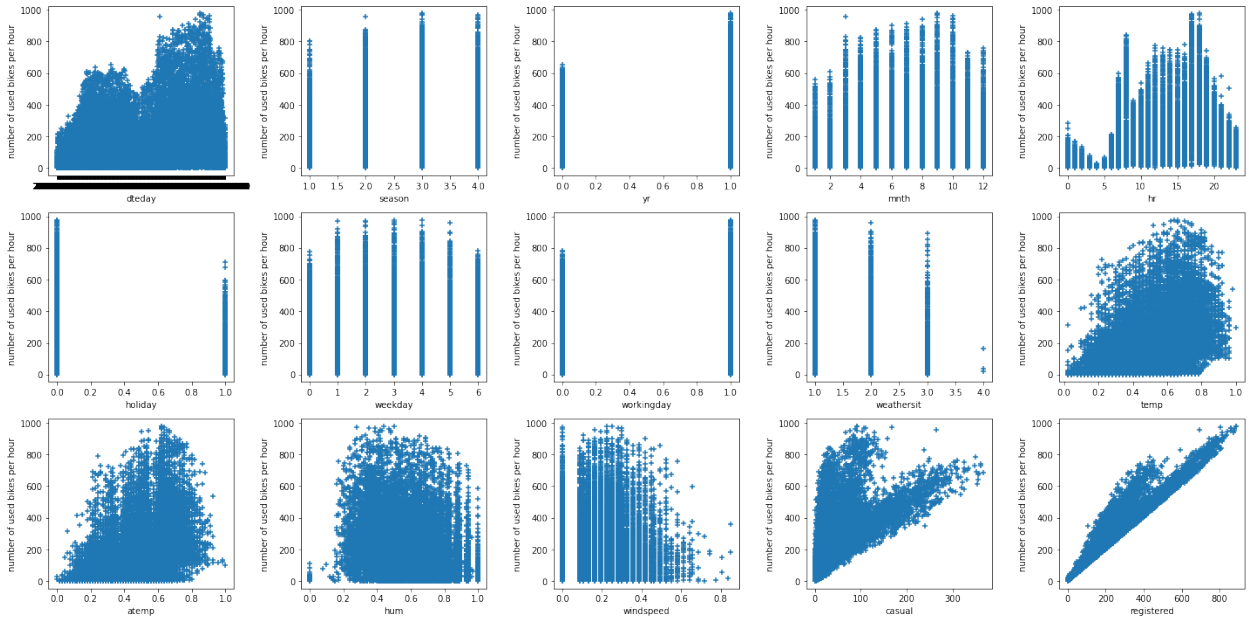
There are 17379 data points and each of them has 17 values with the first value as its record index.

The dimension of the data set is 17379 rows x 17 columns.

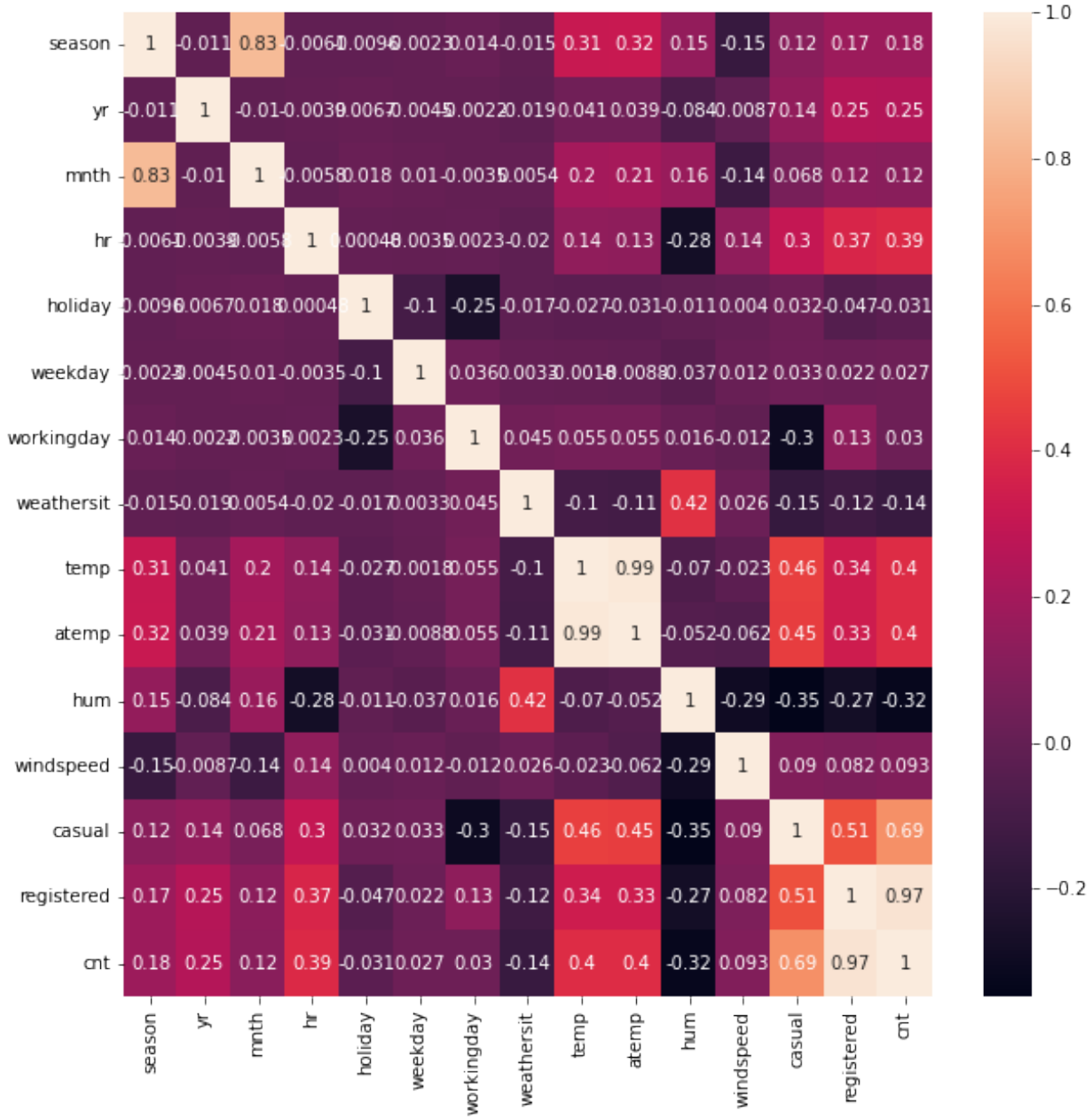
Here is a table of used data types for each column/variable.

instant	int64
dteday	object
season	int64
yr	int64
mnth	int64
hr	int64
holiday	int64
weekday	int64
workingday	int64
weathersit	int64
temp	float64
atemp	float64
hum	float64
windspeed	float64
casual	int64
registered	int64
cnt	int64

(c)



(d)



As the above colored correlation matrix shows,

Feature “registered” is the most positively correlated with the target column cnt.

Feature “hum” is the most negatively correlated with the target column cnt.

Feature “weekday” is the least correlated with the target column cnt.

(e) and (f)

Please refer to my uploaded file CSC2515_HW2.ipynb on MarkUs.

Q3.2 Regression implementations

(a)

Please refer to my uploaded file CSC2515_HW2.ipynb on MarkUs.

(b)

The R^2 score for the ordinary least squares regression model: 0.376869515244005.

(c)

Please refer to my uploaded file CSC2515_HW2.ipynb on MarkUs.

(d)

The R^2 score for the ordinary least squares regression model with the new data: 0.6817966649545371.

(e)

Please refer to my uploaded file CSC2515_HW2.ipynb on MarkUs.

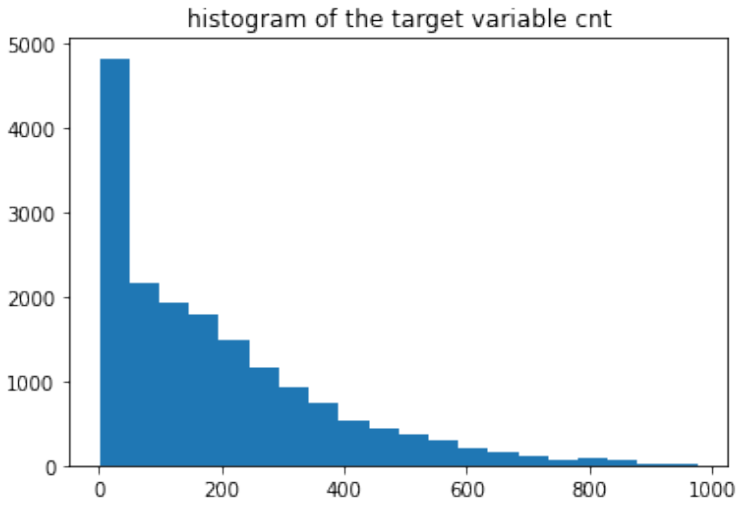
(f)

The R^2 score for the locally weighted regression model with $\tau = 1$: 0.8400597127148324.

The R^2 score for the locally weighted regression model with $\tau = 0.00001$: 0.639203438267656.
This R^2 score on test data is lower than that for the ordinary least squares regression model obtained in part (d), which means this locally weighted regression model has worse generalized performance (on unseen data). This verifies that as $\tau \to 0$, it will behave like a K-NN classifier with K = 1.

The R^2 score for the locally weighted regression model with $\tau = 10000$: 0.687948532428465.
This R^2 score is very close to that for the ordinary least squares regression model obtained in part (d). Thus this verifies that as $\tau \to \infty$, the weight for each query x will be the same, which behaves like an ordinary linear regression.

(g)



It seems to follow a Poisson distribution.

(h)

Please refer to my uploaded file CSC2515_HW2.ipynb on MarkUs.

(i)

The D score for the Poisson regression model with power=1: 0.8011623112319634.

(j)

Here is a table for final weights for the Linear Regression and Poisson Regression.

index	Linear Regression	Poisson Regression
bias term	20.862634324386363	1.924952964682334
yr	83.95015855354656	0.4645300481458441
holiday	-16.405679805095247	0.2875040869857316
workingday	16.239954305477454	0.47085137118702397
temp	258.880303749871	1.162803043967138
hum	-88.76921632573054	-0.22172087942452604
windspeed	-28.827207723702003	-0.11109139227321514
season_1	-30.68178659380021	0.22910035951758667
season_2	12.366716971514279	0.5352619090702629
season_3	2.5458419534887753	0.49315694430423473
season_4	36.63186199318034	0.6974337517902814
mnth_1	3.140150996583259	0.06577187460078149
mnth_2	4.010237706870072	0.17162925356708195
mnth_3	10.061381033765286	0.23255914035089342
mnth_4	0.6103158755275047	0.1940759215376198
mnth_5	10.93950716147998	0.22592140875206942
mnth_6	-7.166016996536484	0.15875895942039467
mnth_7	-21.828706963031607	0.10233130062489523
mnth_8	-3.4844796517955388	0.17499917484093874
mnth_9	26.518933390958182	0.30089000719585857
mnth_10	11.716204294982958	0.20425660836666784
mnth_11	-10.097890724153586	0.10135517907594226
mnth_12	-3.5570018002543975	0.10240413634939587
hr_0	-120.87674215999424	-0.6720421760312077
hr_1	-139.06714585296612	-1.13221131245231
hr_2	-148.45460156926532	-1.5442880506412726
hr_3	-156.67667170254725	-2.1076113963220253
hr_4	-159.69337306835135	-2.50688224195138
hr_5	-142.43987977009556	-1.6180557349896387
hr_6	-84.28057816210348	-0.2857160580137153
hr_7	53.059313731392834	0.7569424967391889
hr_8	182.13605621872975	1.2154120125402048
hr_9	41.10928869888892	0.7047692431741233
hr_10	-14.488468775270036	0.4301736522528837
hr_11	12.098436462503614	0.5803841934641278
hr_12	44.812757508729874	0.726681870519695
hr_13	40.98577609757554	0.7156393978515975
hr_14	20.787333103425354	0.6336483997276838
hr_15	40.70195348263249	0.7194911757499327
hr_16	96.70550286318289	0.91687227686169
hr_17	254.69868054375934	1.329559485781656

index	Linear Regression	Poisson Regression
hr_18	222.91320725516618	1.2674651011721723
hr_19	109.29207238005245	0.9725329999799837
hr_20	30.478767526699755	0.6740272359348881
hr_21	-17.262879531248757	0.42070655754827424
hr_22	-51.8251018042635	0.15928736230372909
hr_23	-93.85106915223855	-0.2018335265176887
weekday_0	5.1589002404277835	0.5719899317908493
weekday_1	-2.0605289275402754	0.14329162435231402
weekday_2	-2.924393019797691	0.14134308941456392
weekday_3	3.004081677310154	0.1647794676471348
weekday_4	-0.5080300432909723	0.16058706641388745
weekday_5	2.32314481369869	0.17835421034469218
weekday_6	15.869459583597745	0.6246075747190463
weathersit_1	44.66947584496211	0.8326639344280568
weathersit_2	33.4335402757024	0.7715049997903376
weathersit_3	-19.038403597456437	0.3447065236998056
weathersit_4	-38.20197819869348	0.006077506764346504

The most significant feature for Linear Regression is "temp" with weight of 258.880303749871. Since riding bike is highly depends on the outdoor temperature. As the normalized temperature increases, the expected number of total rental bikes tend to increase.

The least significant feature for Linear Regression is "weekday_4" with weight of -0.5080300432909723. As "weekday_4" equals to 1 when it is the fifth day of the week, which is Friday. It has the least contribution to the changes in the expected number of total rental bikes. Maybe this is because that less people rent bikes on Friday compared to other days of the week.

The most significant feature for Poisson Regression: "hr_4" with weight of -2.50688224195138. As "hr_4" equals to 1 when it is 5am, the total rental bikes tend to be fewer than other hours, having the greatest effect to the expected number of total rental bikes.

The least significant feature for Poisson Regression: "weathersit_4" with weight of 0.006077506764346504. As "weathersit_4" equals to 1 when there is an extreme weather, which might have a very small effect on the total number of rental bikes.