

High-Dimensional Statistics and Generalized Linear Models

TrungTin Nguyen

STATIFY team, Inria centre at the University Grenoble Alpes, France



LABORATOIRE
JEAN KUNTZMANN
MATHÉMATIQUES APPLIQUÉES - INFORMATIQUE



Statistical Analysis and Document Mining

Complementary Course, Master of Applied Mathematics in Grenoble

- 1 Previous Episode: Discriminative Approaches for Classification Problems
 - Multiple logistic regression
 - Nearest-neighbour methods
 - High-dimensional data classification
- 2 Curse of Dimensionality
 - A synthetically generated Oil Flow data set
 - KNN learning algorithm
 - Curse of dimensionality for kNN
 - Lost in the immensity of high-dimensional spaces
 - The impact of high dimensionality on statistics is multiple!
- 3 Generalized Linear Models
 - Bikeshare data set: motivation example
 - Bikeshare data: linear regression
 - Bikeshare data: Poisson regression
 - Bikeshare data: linear regression and Poisson regression
 - Generalized linear models

- 1 Previous Episode: Discriminative Approaches for Classification Problems
 - Multiple logistic regression
 - Nearest-neighbour methods
 - High-dimensional data classification
- 2 Curse of Dimensionality
 - A synthetically generated Oil Flow data set
 - KNN learning algorithm
 - Curse of dimensionality for kNN
 - Lost in the immensity of high-dimensional spaces
 - The impact of high dimensionality on statistics is multiple!
- 3 Generalized Linear Models
 - Bikeshare data set: motivation example
 - Bikeshare data: linear regression
 - Bikeshare data: Poisson regression
 - Bikeshare data: linear regression and Poisson regression
 - Generalized linear models

Previous episode: multiple logistic regression

- ✍ We are given a **training dataset** $\mathcal{D} \equiv \{(\mathbf{x}_n, y_n)\}_{n \in [N]}$, $y_{[N]} \in [2] \equiv \mathcal{C}$, i.i.d. sampled from **the true (but unknown) joint PDF** of (\mathbf{X}, Y) .
- ❓ How to model relationship between $p_c(\mathbf{X}) = \mathbb{P}(Y = c|\mathbf{X})$, $c \in \mathcal{C}$, and \mathbf{X} ?
- ✍ **Multiple logistic regression** (MLR) takes the form $p_2(\mathbf{X}) = 1 - p_1(\mathbf{X})$, and models the probability that Y belongs to a particular category instead Y ,

$$\log\left(\frac{p_1(\mathbf{X})}{1 - p_1(\mathbf{X})}\right) = \beta_0 + \sum_{p=1}^P \beta_p X_p, \text{ or } p_1(\mathbf{X}) = \frac{\exp(\beta_0 + \sum_{p=1}^P \beta_p X_p)}{1 + \exp(\beta_0 + \sum_{p=1}^P \beta_p X_p)}.$$

- ✍ **Training error:** using non-linear LS or maximum likelihood estimation (MLE), we obtain $\hat{\beta}$ and $\hat{r}_{\mathcal{D}}(\mathbf{x}_n) = \operatorname{argmax}_{c \in \mathcal{C}} p_c(\mathbf{x}_n)$ such that $\forall n \in [N]$,

$$y_n \approx \hat{r}_{\mathcal{D}}(\mathbf{x}_n), \text{ or equivalent, } \mathcal{L}(\hat{r}_{\mathcal{D}}, \mathcal{D}) \equiv \frac{1}{N} \sum_{n=1}^N \mathbb{1}[y_n \neq \hat{r}_{\mathcal{D}}(\mathbf{x}_n)] \approx 0. \quad (1)$$

- ❓ **Test (generalization) error:** for any **new sample** (\mathbf{x}^*, y^*) , how we guarantee

$$y^* \approx \hat{r}_{\mathcal{D}}(\mathbf{x}^*), \text{ or equivalent, } \mathcal{L}(\hat{r}_{\mathcal{D}}) \equiv \mathbb{E}_{\mathbf{X}, Y} [\mathbb{1}(Y \neq \hat{r}_{\mathcal{D}}(\mathbf{X}))] \approx 0? \quad (2)$$

Previous episode: Cross-Validation for MLR

- 1 Split the training dataset randomly into K folds so that we have $\mathcal{D}_1 \cup \dots \cup \mathcal{D}_K = \mathcal{D}$, where \mathcal{D}_k denotes the indices of the observations in part k . There are N_k observations in part k : if N is a multiple of K , then $N_k = N/K$.
- 2 For each $k \in [K]$, fit a model $\hat{r}^{(-k)}(\mathbf{x}, \mathbf{m})$, indexed by a tuning parameter (or a model) $\mathbf{m} \in \mathcal{M}$, on all samples from the training set except those in the k th fold.
- 3 Estimating **test error** $\mathcal{L}(\hat{r}_{\mathcal{D}})$ via averaging the final resulting MSE estimates

$$\mathcal{L}(\hat{r}_{\mathcal{D}}) \approx \underbrace{\frac{1}{K} \sum_{k=1}^K}_{\text{average over } K \text{ folds}} \underbrace{\left[\frac{1}{N_k} \sum_{n \in \mathcal{D}_k} \mathbb{1} \left(y_n \neq \hat{r}^{(-k)}(\mathbf{x}_n) \right) \right]}_{\text{Estimate test error for each fold}} \equiv \text{CV}(\hat{r}, K, \mathbf{m}).$$

- 4 **Best data-driven model:** $\hat{\mathbf{m}} \equiv \operatorname{argmin}_{\mathbf{m} \in \mathcal{M}} \text{CV}(\hat{r}, K, \mathbf{m})$.
 $K = ?$ Setting $K = N$ yields N -fold or leave-one out cross-validation (LOOCV, **high variance**). A common better choice $K = 5$ or 10 .

- 1 Previous Episode: Discriminative Approaches for Classification Problems
 - Multiple logistic regression
 - **Nearest-neighbour methods**
 - High-dimensional data classification
- 2 Curse of Dimensionality
 - A synthetically generated Oil Flow data set
 - KNN learning algorithm
 - Curse of dimensionality for kNN
 - Lost in the immensity of high-dimensional spaces
 - The impact of high dimensionality on statistics is multiple!
- 3 Generalized Linear Models
 - Bikeshare data set: motivation example
 - Bikeshare data: linear regression
 - Bikeshare data: Poisson regression
 - Bikeshare data: linear regression and Poisson regression
 - Generalized linear models

Previous episode: Nearest-neighbour methods

- ✎ We are given a **training dataset** $\mathcal{D} \equiv \{(\mathbf{x}_n, y_n)\}_{n \in [M]}$, $y_{[M]} \in \mathcal{C} \subsetneq \mathbb{N}$, i.i.d. sampled from **the true (but unknown) joint PDF** of (\mathbf{X}, Y) .
- ❓ How to model relationship between $p_c(\mathbf{X}) = \mathbb{P}(Y = c|\mathbf{X})$, $c \in \mathcal{C}$, and \mathbf{X} ?
- ✎ **K-Nearest-neighbours (kNN)** approximate the probability that Y belongs to a particular category instead Y ,

$$\mathbb{P}(Y = c|\mathbf{X} = \mathbf{x}) \approx \frac{1}{K} \sum_{\mathbf{x}_n \in \mathcal{N}_K(\mathbf{x})} \mathbb{1}_{y=c}(y_n).$$

- kNN is a **nonparametric classifier**: it makes **no assumptions about the data**. \rightarrow It can be **very flexible** but may **not be optimal** when we **know something about the data**.
 - The hyper-parameter K usually chosen via cross-validation.
- ❓ **It works well in low dimensions, but suffers from the curse of dimensionality.** To be verified soon!

- 1 Previous Episode: Discriminative Approaches for Classification Problems
 - Multiple logistic regression
 - Nearest-neighbour methods
 - **High-dimensional data classification**
- 2 Curse of Dimensionality
 - A synthetically generated Oil Flow data set
 - KNN learning algorithm
 - Curse of dimensionality for kNN
 - Lost in the immensity of high-dimensional spaces
 - The impact of high dimensionality on statistics is multiple!
- 3 Generalized Linear Models
 - Bikeshare data set: motivation example
 - Bikeshare data: linear regression
 - Bikeshare data: Poisson regression
 - Bikeshare data: linear regression and Poisson regression
 - Generalized linear models

High-dimensional data classification

- ✎ We are given a **training dataset** $\mathcal{D} \equiv \{(\mathbf{x}_n, y_n)\}_{n \in [N]}$, $y_{[N]} \in \mathcal{C}$, i.i.d. sampled from **the true (but unknown) joint PDF** of (\mathbf{X}, Y) .
- 👤 In genomics and other areas of computational biology, the number of features P is much larger than the number of observations N , often written as $P \gg N$, where $\mathbf{x}_n \in \mathbb{R}^P$.
- ❓ How to model relationship between $p_c(\mathbf{X}) = \mathbb{P}(Y = c|\mathbf{X})$, $c \in \mathcal{C}$, and \mathbf{X} ?
- ❓ Which is suitable for high-dimensional data: **Discriminative approaches (CM5)** or **Generative approaches (CM6)**.
 - ① K-nearest neighbors (K-NN) (CM5),
 - ② Logistic Regression (CM5),
 - ③ Linear Discriminant Analysis (CM5),
 - ④ Naive Bayes classifier (CM6).

- 1 Previous Episode: Discriminative Approaches for Classification Problems
 - Multiple logistic regression
 - Nearest-neighbour methods
 - High-dimensional data classification
- 2 Curse of Dimensionality
 - A synthetically generated Oil Flow data set
 - KNN learning algorithm
 - Curse of dimensionality for kNN
 - Lost in the immensity of high-dimensional spaces
 - The impact of high dimensionality on statistics is multiple!
- 3 Generalized Linear Models
 - Bikeshare data set: motivation example
 - Bikeshare data: linear regression
 - Bikeshare data: Poisson regression
 - Bikeshare data: linear regression and Poisson regression
 - Generalized linear models

Outline

- 1 Previous Episode: Discriminative Approaches for Classification Problems
 - Multiple logistic regression
 - Nearest-neighbour methods
 - High-dimensional data classification
- 2 Curse of Dimensionality
 - A synthetically generated Oil Flow data set
 - KNN learning algorithm
 - Curse of dimensionality for kNN
 - Lost in the immensity of high-dimensional spaces
 - The impact of high dimensionality on statistics is multiple!
- 3 Generalized Linear Models
 - Bikeshare data set: motivation example
 - Bikeshare data: linear regression
 - Bikeshare data: Poisson regression
 - Bikeshare data: linear regression and Poisson regression
 - Generalized linear models

A synthetically generated Oil Flow data set¹

- Representing $N = 1000$ **data points** (for each training, validation, and test sets) taken from a pipeline containing a mixture of oil, water, and gas.
- Comprising a $P = 12$ -**dimensional input vector** consisting of measurements taken with gamma ray densitometers that measure the attenuation of gamma rays passing along narrow beams through the pipe.
- Each data point is labelled according to which of the three geometrical classes it belongs to: $Y \in \{\text{Stratified}, \text{Annular}, \text{Homogeneous}\}$, see Figure 1 for more details.

¹Bishop, C. M. and G. D. James (1993). *Analysis of multiphase flows using dual-energy gamma densitometry and neural networks*. Nuclear Instruments and Methods in Physics Research.

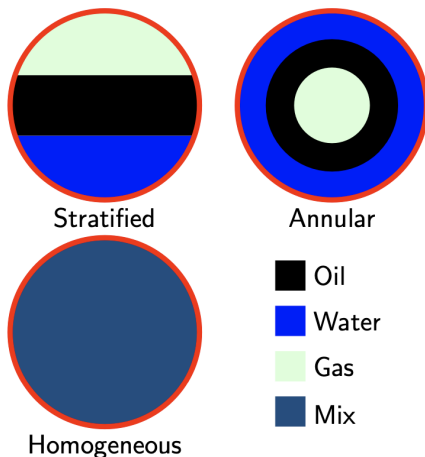


Figure 1: The three geometrical configurations of the oil, water, and gas phases used to generate the oil-flow data set. For each configuration, the proportions of the three phases can vary [Bishop, 2006, Figure A.2].

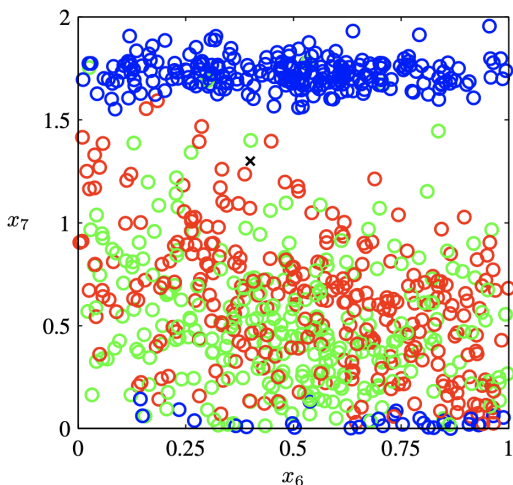
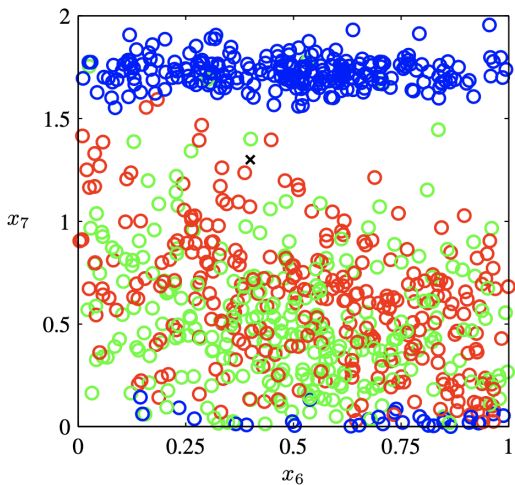
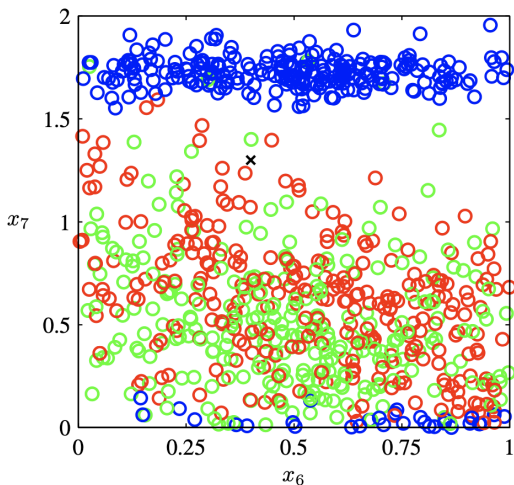


Figure 2: Scatter plot of the oil flow data for input variables x_6 and x_7 , in which red denotes the 'homogenous' class, green denotes the 'annular' class, and blue denotes the 'laminar' class. Our goal is to classify the new test point denoted by 'x'. [Bishop, 2006, Figure 1.19].



? Your intuitive solution ...

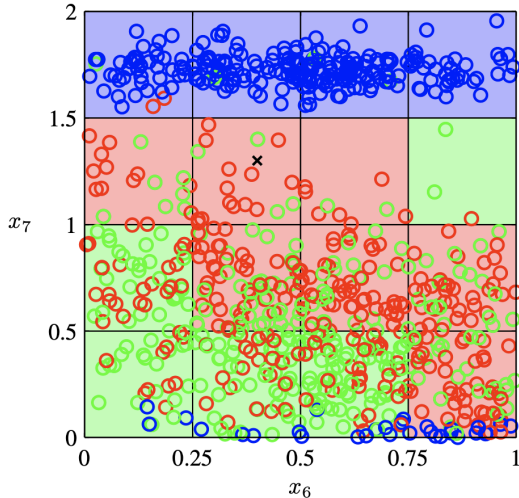
 New test point denoted by 'x' should be x, x or x?



Intuitive solution the cross \mathbf{x} is surrounded by numerous red points, and so we might suppose that it belongs to the red class. There are also plenty of green points nearby, so we might think that it could instead belong to the green class. It seems unlikely that it belongs to the blue class.

Outline

- 1 Previous Episode: Discriminative Approaches for Classification Problems
 - Multiple logistic regression
 - Nearest-neighbour methods
 - High-dimensional data classification
- 2 **Curse of Dimensionality**
 - A synthetically generated Oil Flow data set
 - **KNN learning algorithm**
 - Curse of dimensionality for kNN
 - Lost in the immensity of high-dimensional spaces
 - The impact of high dimensionality on statistics is multiple!
- 3 Generalized Linear Models
 - Bikeshare data set: motivation example
 - Bikeshare data: linear regression
 - Bikeshare data: Poisson regression
 - Bikeshare data: linear regression and Poisson regression
 - Generalized linear models



KNN learning algorithm: divide the input space into regular cells and any new test point is assigned to the class that has a **majority number of representatives in the same cell as the test point**. [Bishop, 2006, Figure 1.20]. ? A few potentially serious shortcomings ...

- 1 Previous Episode: Discriminative Approaches for Classification Problems
 - Multiple logistic regression
 - Nearest-neighbour methods
 - High-dimensional data classification
- 2 **Curse of Dimensionality**
 - A synthetically generated Oil Flow data set
 - KNN learning algorithm
 - **Curse of dimensionality for kNN**
 - Lost in the immensity of high-dimensional spaces
 - The impact of high dimensionality on statistics is multiple!
- 3 Generalized Linear Models
 - Bikeshare data set: motivation example
 - Bikeshare data: linear regression
 - Bikeshare data: Poisson regression
 - Bikeshare data: linear regression and Poisson regression
 - Generalized linear models



$$D = 1$$

Illustration of the curse of dimensionality: number of cells = $3^1 = 3$
[[Bishop, 2006](#), Figure 1.21].

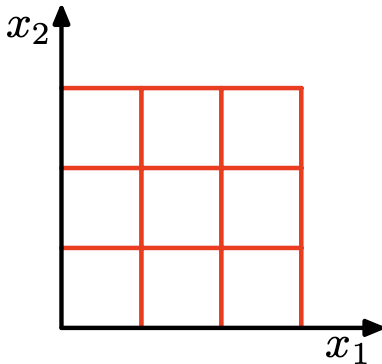


Illustration of the curse of dimensionality: when $P = 2$, number of cells $= 3^2 = 9$ [Bishop, 2006, Figure 1.21].

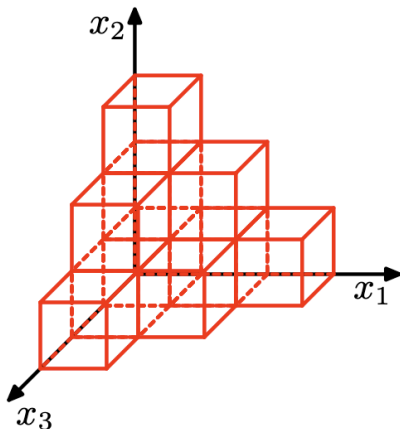


Illustration of the curse of dimensionality: when $P = 3$, number of cells $= 3^3 = 27$ [Bishop, 2006, Figure 1.21].

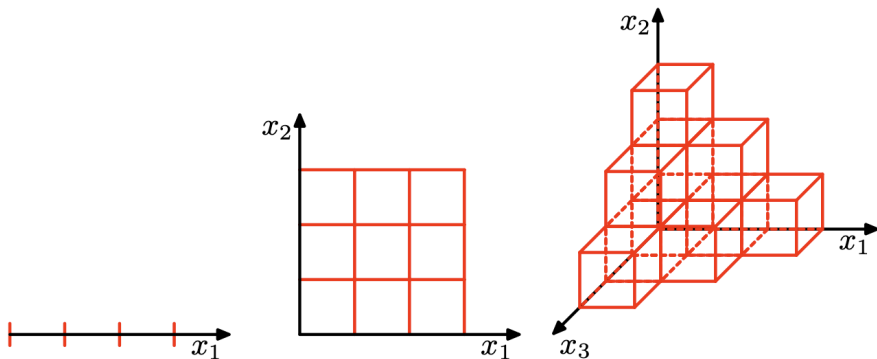


Illustration of the curse of dimensionality: exponentially large number of cells 3^P , e.g., $3^{12} = 531441$ is that we would need an exponentially large quantity of training data in order to ensure that the cells are not empty [Bishop, 2006, Figure 1.21].

Outline

- 1 Previous Episode: Discriminative Approaches for Classification Problems
 - Multiple logistic regression
 - Nearest-neighbour methods
 - High-dimensional data classification
- 2 **Curse of Dimensionality**
 - A synthetically generated Oil Flow data set
 - KNN learning algorithm
 - Curse of dimensionality for kNN
 - **Lost in the immensity of high-dimensional spaces**
 - The impact of high dimensionality on statistics is multiple!
- 3 Generalized Linear Models
 - Bikeshare data set: motivation example
 - Bikeshare data: linear regression
 - Bikeshare data: Poisson regression
 - Bikeshare data: linear regression and Poisson regression
 - Generalized linear models

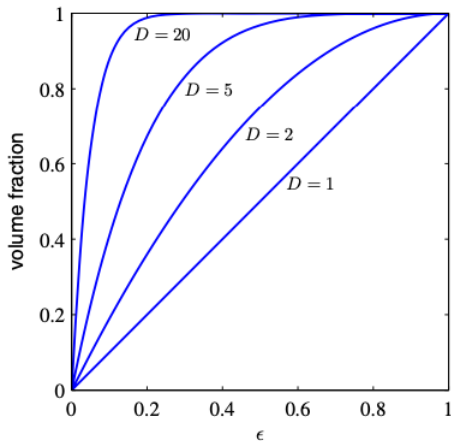
Lost in the immensity of high-dimensional spaces

☞ Our geometrical intuitions, formed through a life spent in a space of three dimensions, can fail badly when we consider spaces of higher dimensionality [Bishop, 2006].

✿ Fraction of the volume of the sphere that lies between radius $r = 1 - \epsilon$ and $r = 1$ is given by

$$\frac{V_P(1) - V_P(1 - \epsilon)}{V_P(1)} = 1 - (1 - \epsilon)^P \longrightarrow 1 \text{ for large } P \text{ and small } \epsilon!$$

Here, $V_P(1)$ is a volume of a sphere of radius r in $P(\equiv D)$ dimensions.



Plot of the fraction of the volume of a sphere that lies between radius $r = 1 - \epsilon$ and $r = 1$ for various values of the dimensionality $P(\equiv D)$

[Bishop, 2006, Figure 1.22]. ANY comments ?

👉 In spaces of high dimensionality, most of the volume of a sphere is concentrated in a thin shell near the surface!

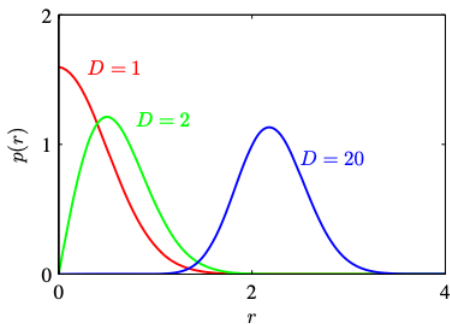
Lost in the immensity of high-dimensional spaces

☞ Our geometrical intuitions, formed through a life spent in a space of three dimensions, can fail badly when we consider spaces of higher dimensionality [Bishop, 2006].

⚙ Behaviour of a Gaussian distribution in a high-dimensional space.

☞ If we transform from Cartesian to polar coordinates, and then integrate out the directional variables, we obtain an expression for the density $p(r)$ as a function of radius r from the origin.

☞ Thus $p(r)\delta r$ is the probability mass inside a thin shell of thickness δr located at radius r .



Plot of the probability density with respect to radius r of a Gaussian distribution for various values of the dimensionality $P(\equiv D)$.

[Bishop, 2006, Figure 1.23]. ANY comments ?

👉 For large $P(\equiv D)$ most of the probability mass of a Gaussian is concentrated in a thin shell!

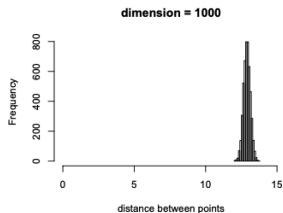
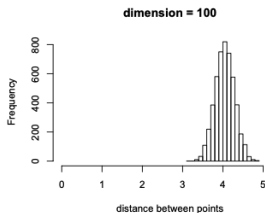
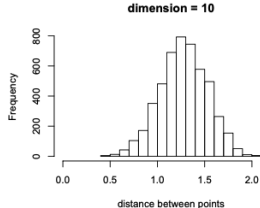
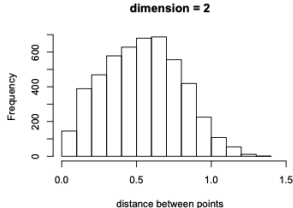
Lost in the immensity of high-dimensional spaces

☞ High-dimensional spaces are vast and data points are isolated in their immensity [Giraud, 2021].

- So far, we have considered the situation where we want to explain a response variable $Y \in \mathbb{R}$ by P variables $X_1, \dots, X_P \in [0, 1]$. Assume that each variable X_p follows a uniform distribution on $[0, 1]$.
- If these variables are independent, then the variable $X = (X_1, \dots, X_P) \in [0, 1]^P$ follows a uniform distribution on the hypercube $[0, 1]^P$.
- Consider classical regression, given $\epsilon_{[N]}$ independent and centered,

$$Y_n = f(\mathbf{X}_n) + \epsilon_n, \quad n \in [N] \text{ with } f : [0, 1]^P \rightarrow \mathbb{R}. \quad (3)$$

➕ Assuming that the function f is smooth, it is natural to estimate $f(x)$ by some average of the Y_n associated to the \mathbf{X}_n in the vicinity of x . The most simple version of this idea is the kNN estimator, where $f(x)$ is estimated by the mean of the Y_n associated to the k points \mathbf{X}_n , which are the nearest from x !



Histograms of the pairwise-distances between $N = 100$ points sampled uniformly in the hypercube $[0, 1]^P$, for $P = 2, 10, 100$, and 1000 [Giraud, 2021, Figure 1.3].

ANY comments ?

👉 For large P : (1) the minimal distance between two points increases, (2) all the points are at a similar distance from the others, so the notion of “nearest points” vanishes.

- 1 Previous Episode: Discriminative Approaches for Classification Problems
 - Multiple logistic regression
 - Nearest-neighbour methods
 - High-dimensional data classification
- 2 **Curse of Dimensionality**
 - A synthetically generated Oil Flow data set
 - KNN learning algorithm
 - Curse of dimensionality for kNN
 - Lost in the immensity of high-dimensional spaces
 - **The impact of high dimensionality on statistics is multiple!**
- 3 Generalized Linear Models
 - Bikeshare data set: motivation example
 - Bikeshare data: linear regression
 - Bikeshare data: Poisson regression
 - Bikeshare data: linear regression and Poisson regression
 - Generalized linear models

- ① High-dimensional spaces are vast and data points are isolated in their immensity.
- ② The accumulation of small fluctuations in many different directions can produce a large global fluctuation.
- ③ An event that is an accumulation of rare events may not be rare.
- ④ Numerical computations and optimizations in high-dimensional spaces can be overly intensive.

Outline

- 1 Previous Episode: Discriminative Approaches for Classification Problems
 - Multiple logistic regression
 - Nearest-neighbour methods
 - High-dimensional data classification
- 2 Curse of Dimensionality
 - A synthetically generated Oil Flow data set
 - KNN learning algorithm
 - Curse of dimensionality for kNN
 - Lost in the immensity of high-dimensional spaces
 - The impact of high dimensionality on statistics is multiple!
- 3 Generalized Linear Models
 - Bikeshare data set: motivation example
 - Bikeshare data: linear regression
 - Bikeshare data: Poisson regression
 - Bikeshare data: linear regression and Poisson regression
 - Generalized linear models

- 1 Previous Episode: Discriminative Approaches for Classification Problems
 - Multiple logistic regression
 - Nearest-neighbour methods
 - High-dimensional data classification
- 2 Curse of Dimensionality
 - A synthetically generated Oil Flow data set
 - KNN learning algorithm
 - Curse of dimensionality for kNN
 - Lost in the immensity of high-dimensional spaces
 - The impact of high dimensionality on statistics is multiple!
- 3 Generalized Linear Models
 - Bikeshare data set: motivation example
 - Bikeshare data: linear regression
 - Bikeshare data: Poisson regression
 - Bikeshare data: linear regression and Poisson regression
 - Generalized linear models

Bikeshare data set: motivation example

☞ We consider the **Bikeshare** data set. The response is **bikers**, the **number of hourly users of a bike sharing** program in Washington, DC. This response value is **neither qualitative nor quantitative**: instead, it **takes on non-negative integer values, or counts**.

⊕ Predicting **bikers** using the covariates **mnth** (month of the year), **hr** (hour of the day, from 0 to 23), **workingday** (an indicator variable that equals 1 if it is neither a weekend nor a holiday), **temp** (the normalized temperature, in Celsius), and **weathersit** (a qualitative variable that takes on one of four possible values: clear; misty or cloudy; light rain or light snow; or heavy rain or heavy snow.)

```
> head(Bikeshare)
```

	season	mnth	day	hr	holiday	weekday	workingday	weathersit	temp	atemp	hum	windspeed	casual	registered	bikers
1	1	Jan	1	0	0	6	0	clear	0.24	0.2879	0.81	0.0000	3	13	16
2	1	Jan	1	1	0	6	0	clear	0.22	0.2727	0.80	0.0000	8	32	40
3	1	Jan	1	2	0	6	0	clear	0.22	0.2727	0.80	0.0000	5	27	32
4	1	Jan	1	3	0	6	0	clear	0.24	0.2879	0.75	0.0000	3	10	13
5	1	Jan	1	4	0	6	0	clear	0.24	0.2879	0.75	0.0000	0	1	1
6	1	Jan	1	5	0	6	0	cloudy/misty	0.24	0.2576	0.75	0.0896	0	1	1

Outline

- 1 Previous Episode: Discriminative Approaches for Classification Problems
 - Multiple logistic regression
 - Nearest-neighbour methods
 - High-dimensional data classification
- 2 Curse of Dimensionality
 - A synthetically generated Oil Flow data set
 - KNN learning algorithm
 - Curse of dimensionality for kNN
 - Lost in the immensity of high-dimensional spaces
 - The impact of high dimensionality on statistics is multiple!
- 3 Generalized Linear Models
 - Bikeshare data set: motivation example
 - **Bikeshare data: linear regression**
 - Bikeshare data: Poisson regression
 - Bikeshare data: linear regression and Poisson regression
 - Generalized linear models

```
> mod.lm2 <- lm(
+   bikers ~ mnth + hr + workingday + temp + weathersit,
+   data = Bikeshare
+ )
> summary(mod.lm2)
```

Call:

```
lm(formula = bikers ~ mnth + hr + workingday + temp + weathersit,
    data = Bikeshare)
```

Residuals:

Min	1Q	Median	3Q	Max
-299.00	-45.70	-6.23	41.08	425.29

Coefficients:

	Estimate	Std. Error	t value	Pr(> t)	
(Intercept)	73.5974	5.1322	14.340	< 2e-16	***
mnth1	-46.0871	4.0855	-11.281	< 2e-16	***
mnth2	-39.2419	3.5391	-11.088	< 2e-16	***
mnth3	-29.5357	3.1552	-9.361	< 2e-16	***
mnth4	-4.6622	2.7406	-1.701	0.08895	.
mnth5	26.4700	2.8508	9.285	< 2e-16	***
mnth6	21.7317	3.4651	6.272	3.75e-10	***
mnth7	-0.7626	3.9084	-0.195	0.84530	
mnth8	7.1560	3.5347	2.024	0.04295	*
mnth9	20.5912	3.0456	6.761	1.46e-11	***
mnth10	29.7472	2.6995	11.019	< 2e-16	***
mnth11	14.2229	2.8604	4.972	6.74e-07	***

mnth11	14.2229	2.8604	4.972	6.74e-07	***
hr1	-96.1420	3.9554	-24.307	< 2e-16	***
hr2	-110.7213	3.9662	-27.916	< 2e-16	***
hr3	-117.7212	4.0165	-29.310	< 2e-16	***
hr4	-127.2828	4.0808	-31.191	< 2e-16	***
hr5	-133.0495	4.1168	-32.319	< 2e-16	***
hr6	-120.2775	4.0370	-29.794	< 2e-16	***
hr7	-75.5424	3.9916	-18.925	< 2e-16	***
hr8	23.9511	3.9686	6.035	1.65e-09	***
hr9	127.5199	3.9500	32.284	< 2e-16	***
hr10	24.4399	3.9360	6.209	5.57e-10	***
hr11	-12.3407	3.9361	-3.135	0.00172	**
hr12	9.2814	3.9447	2.353	0.01865	*
hr13	41.1417	3.9571	10.397	< 2e-16	***
hr14	39.8939	3.9750	10.036	< 2e-16	***
hr15	30.4940	3.9910	7.641	2.39e-14	***
hr16	35.9445	3.9949	8.998	< 2e-16	***
hr17	82.3786	3.9883	20.655	< 2e-16	***
hr18	200.1249	3.9638	50.488	< 2e-16	***
hr19	173.2989	3.9561	43.806	< 2e-16	***
hr20	90.1138	3.9400	22.872	< 2e-16	***
hr21	29.4071	3.9362	7.471	8.74e-14	***
hr22	-8.5883	3.9332	-2.184	0.02902	*
hr23	-37.0194	3.9344	-9.409	< 2e-16	***
workingday	1.2696	1.7845	0.711	0.47681	
temp	157.2094	10.2612	15.321	< 2e-16	***
weathersitcloudy/misty	-12.8903	1.9643	-6.562	5.60e-11	***
weathersitlight rain/snow	-66.4944	2.9652	-22.425	< 2e-16	***

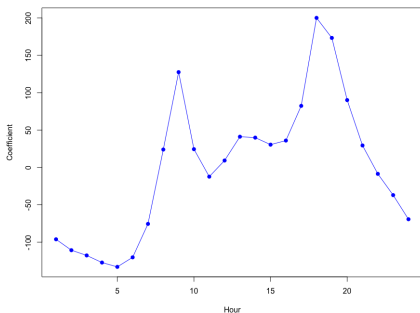
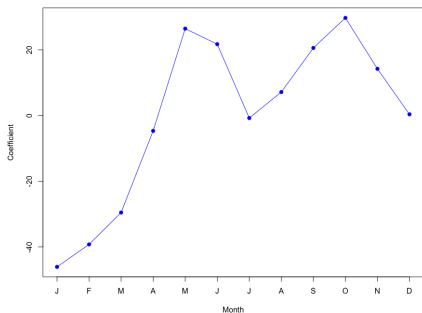
hr20	90.1138	3.9400	22.872	< 2e-16	***
hr21	29.4071	3.9362	7.471	8.74e-14	***
hr22	-8.5883	3.9332	-2.184	0.02902	*
hr23	-37.0194	3.9344	-9.409	< 2e-16	***
workingday	1.2696	1.7845	0.711	0.47681	
temp	157.2094	10.2612	15.321	< 2e-16	***
weathersitcloudy/misty	-12.8903	1.9643	-6.562	5.60e-11	***
weathersitlight rain/snow	-66.4944	2.9652	-22.425	< 2e-16	***
weathersitheavy rain/snow	-109.7446	76.6674	-1.431	0.15234	

Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 76.5 on 8605 degrees of freedom

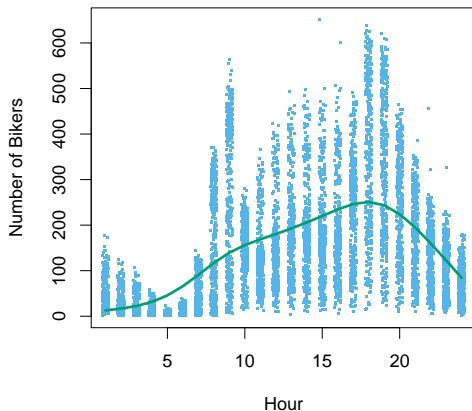
Multiple R-squared: 0.6745, Adjusted R-squared: 0.6731

F-statistic: 457.3 on 39 and 8605 DF, p-value: < 2.2e-16



A least squares linear regression model was fit to predict bikers in the Bikeshare data set. Left: The coefficients associated with the month of the year. Bike usage is highest in the spring and fall, and lowest in the winter. Right: The coefficients associated with the hour of the day. Bike usage is highest during peak commute times, and lowest overnight [James et al., 2021, Figure 4.13]

🤔 At first glance, fitting a linear regression model to the Bikeshare seems to provide reasonable and intuitive results.

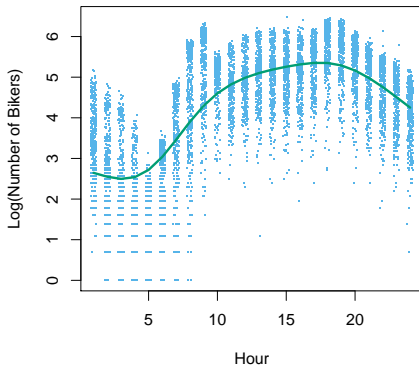
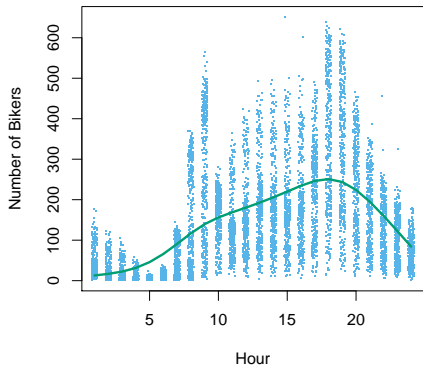


For the most part, as the **mean number of bikers increases**, so does **the variance in the number of bikers** [James et al., 2021, Figure 4.14].

❓ At first glance, fitting a linear regression model to the Bikeshare seems to provide reasonable and intuitive results.

- ① **9.6% of the fitted values** in the Bikeshare data set are **negative**: that is, the linear regression model predicts a **negative number of users during 9.6% of the hours in the data set.**
- ② Since ϵ is a continuous-valued error term, response Y is necessarily **continuous-valued** (quantitative) but the **response bikers** is **integer-valued.**

- 3 The mean-variance relationship is a major violation of the assumptions of a linear model, which state that $Y = \beta_0 + \sum_{p=1}^P X_p \beta_p + \epsilon$, where ϵ is a mean-zero error term with variance σ^2 that is constant, and not a function of the covariates. For the most part, as the mean number of bikers increases, so does the variance in the number of bikers!



Some solutions

- ① Transforming the response Y avoids the possibility of negative predictions, and it overcomes much of the heteroscedasticity in the untransformed data $\log(Y) = \beta_0 + \sum_{p=1}^P X_p \beta_p + \epsilon$.
BUT Making predictions and inference in terms of the log of the response, rather than the response \rightarrow challenges in interpretation!
- ✚ Poisson regression model provides a much more natural and elegant approach for this task! (See more in CC6).
- ⚙ Transforming the mean of response $\mathbb{E}[Y]$ as in the logistic regression!

- 1 Previous Episode: Discriminative Approaches for Classification Problems
 - Multiple logistic regression
 - Nearest-neighbour methods
 - High-dimensional data classification
- 2 Curse of Dimensionality
 - A synthetically generated Oil Flow data set
 - KNN learning algorithm
 - Curse of dimensionality for kNN
 - Lost in the immensity of high-dimensional spaces
 - The impact of high dimensionality on statistics is multiple!
- 3 Generalized Linear Models
 - Bikeshare data set: motivation example
 - Bikeshare data: linear regression
 - **Bikeshare data: Poisson regression**
 - Bikeshare data: linear regression and Poisson regression
 - Generalized linear models

Poisson Regression

- ❓ Definition of Poisson distribution: Suppose that a random variable Y takes on nonnegative integer values, Poisson, *i.e.*, $Y \in \{0, 1, 2, \dots\}$. If Y follows the Poisson distribution, then

$$\mathbb{P}(Y = k) = \frac{e^{-\lambda} \lambda^k}{k!} \text{ for } k = 0, 1, 2, \dots, \lambda > 0. \quad (4)$$

Here, $\mathbb{E}[Y] = \text{var}[Y] = \lambda$.

- ❓ Definition of Poisson regression model:

$$\mathbb{E}[Y] = \lambda \equiv \lambda(X_1, \dots, X_P) = \exp(\beta_0 + \sum_{p=1}^P X_p \beta_p). \quad (5)$$

👉 MLE approach: we want to maximize MLE

$$l(\beta_0, \beta_1, \dots, \beta_P) = \prod_{n=1}^N \frac{e^{-\lambda(x_n)} \lambda(x_n)^{y_n}}{y_n!}, \quad \lambda(x_n) = \exp(\beta_0 + \sum_{p=1}^P x_{np} \beta_p). \quad (6)$$

- 1 Previous Episode: Discriminative Approaches for Classification Problems
 - Multiple logistic regression
 - Nearest-neighbour methods
 - High-dimensional data classification
- 2 Curse of Dimensionality
 - A synthetically generated Oil Flow data set
 - KNN learning algorithm
 - Curse of dimensionality for kNN
 - Lost in the immensity of high-dimensional spaces
 - The impact of high dimensionality on statistics is multiple!
- 3 Generalized Linear Models
 - Bikeshare data set: motivation example
 - Bikeshare data: linear regression
 - Bikeshare data: Poisson regression
 - Bikeshare data: linear regression and Poisson regression
 - Generalized linear models

```
> mod.lm2 <- lm(
+   bikers ~ mnth + hr + workingday + temp + weathersit,
+   data = Bikeshare
+ )
> summary(mod.lm2)
```

Call:

```
lm(formula = bikers ~ mnth + hr + workingday + temp + weathersit,
    data = Bikeshare)
```

Residuals:

Min	1Q	Median	3Q	Max
-299.00	-45.70	-6.23	41.08	425.29

Coefficients:

	Estimate	Std. Error	t value	Pr(> t)	
(Intercept)	73.5974	5.1322	14.340	< 2e-16	***
mnth1	-46.0871	4.0855	-11.281	< 2e-16	***
mnth2	-39.2419	3.5391	-11.088	< 2e-16	***
mnth3	-29.5357	3.1552	-9.361	< 2e-16	***
mnth4	-4.6622	2.7406	-1.701	0.08895	.
mnth5	26.4700	2.8508	9.285	< 2e-16	***
mnth6	21.7317	3.4651	6.272	3.75e-10	***
mnth7	-0.7626	3.9084	-0.195	0.84530	
mnth8	7.1560	3.5347	2.024	0.04295	*
mnth9	20.5912	3.0456	6.761	1.46e-11	***
mnth10	29.7472	2.6995	11.019	< 2e-16	***
mnth11	14.2229	2.8604	4.972	6.74e-07	***


```
> mod.pois <- glm(
+   bikers ~ mnth + hr + workingday + temp + weathersit,
+   data = Bikeshare, family = poisson
+ )
> summary(mod.pois)
```

Call:

```
glm(formula = bikers ~ mnth + hr + workingday + temp + weathersit,
    family = poisson, data = Bikeshare)
```

Deviance Residuals:

Min	1Q	Median	3Q	Max
-20.7574	-3.3441	-0.6549	2.6999	21.9628

Coefficients:

	Estimate	Std. Error	z value	Pr(> z)
(Intercept)	4.118245	0.006021	683.964	< 2e-16 ***
mnth1	-0.670170	0.005907	-113.445	< 2e-16 ***
mnth2	-0.444124	0.004860	-91.379	< 2e-16 ***
mnth3	-0.293733	0.004144	-70.886	< 2e-16 ***
mnth4	0.021523	0.003125	6.888	5.66e-12 ***
mnth5	0.240471	0.002916	82.462	< 2e-16 ***
mnth6	0.223235	0.003554	62.818	< 2e-16 ***
mnth7	0.103617	0.004125	25.121	< 2e-16 ***
mnth8	0.151171	0.003662	41.281	< 2e-16 ***
mnth9	0.233493	0.003102	75.281	< 2e-16 ***
mnth10	0.267573	0.002785	96.091	< 2e-16 ***
mnth11	0.150264	0.003180	47.248	< 2e-16 ***

mnth11	14.2229	2.8604	4.972	6.74e-07	***
hr1	-96.1420	3.9554	-24.307	< 2e-16	***
hr2	-110.7213	3.9662	-27.916	< 2e-16	***
hr3	-117.7212	4.0165	-29.310	< 2e-16	***
hr4	-127.2828	4.0808	-31.191	< 2e-16	***
hr5	-133.0495	4.1168	-32.319	< 2e-16	***
hr6	-120.2775	4.0370	-29.794	< 2e-16	***
hr7	-75.5424	3.9916	-18.925	< 2e-16	***
hr8	23.9511	3.9686	6.035	1.65e-09	***
hr9	127.5199	3.9500	32.284	< 2e-16	***
hr10	24.4399	3.9360	6.209	5.57e-10	***
hr11	-12.3407	3.9361	-3.135	0.00172	**
hr12	9.2814	3.9447	2.353	0.01865	*
hr13	41.1417	3.9571	10.397	< 2e-16	***
hr14	39.8939	3.9750	10.036	< 2e-16	***
hr15	30.4940	3.9910	7.641	2.39e-14	***
hr16	35.9445	3.9949	8.998	< 2e-16	***
hr17	82.3786	3.9883	20.655	< 2e-16	***
hr18	200.1249	3.9638	50.488	< 2e-16	***
hr19	173.2989	3.9561	43.806	< 2e-16	***
hr20	90.1138	3.9400	22.872	< 2e-16	***
hr21	29.4071	3.9362	7.471	8.74e-14	***
hr22	-8.5883	3.9332	-2.184	0.02902	*
hr23	-37.0194	3.9344	-9.409	< 2e-16	***
workingday	1.2696	1.7845	0.711	0.47681	
temp	157.2094	10.2612	15.321	< 2e-16	***
weathersitcloudy/misty	-12.8903	1.9643	-6.562	5.60e-11	***
weathersitlight rain/snow	-66.4944	2.9652	-22.425	< 2e-16	***

hr11	0.336852	0.004720	71.372	< 2e-16	***
hr12	0.494121	0.004392	112.494	< 2e-16	***
hr13	0.679642	0.004069	167.040	< 2e-16	***
hr14	0.673565	0.004089	164.722	< 2e-16	***
hr15	0.624910	0.004178	149.570	< 2e-16	***
hr16	0.653763	0.004132	158.205	< 2e-16	***
hr17	0.874301	0.003784	231.040	< 2e-16	***
hr18	1.294635	0.003254	397.848	< 2e-16	***
hr19	1.212281	0.003321	365.084	< 2e-16	***
hr20	0.914022	0.003700	247.065	< 2e-16	***
hr21	0.616201	0.004191	147.045	< 2e-16	***
hr22	0.364181	0.004659	78.173	< 2e-16	***
hr23	0.117493	0.005225	22.488	< 2e-16	***
workingday	0.014665	0.001955	7.502	6.27e-14	***
temp	0.785292	0.011475	68.434	< 2e-16	***
weathersitcloudy/misty	-0.075231	0.002179	-34.528	< 2e-16	***
weathersitlight rain/snow	-0.575800	0.004058	-141.905	< 2e-16	***
weathersitheavy rain/snow	-0.926287	0.166782	-5.554	2.79e-08	***

Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for poisson family taken to be 1)

Null deviance: 1052921 on 8644 degrees of freedom
 Residual deviance: 228041 on 8605 degrees of freedom
 AIC: 281159

Number of Fisher Scoring iterations: 5

hr20	90.1138	3.9400	22.872	< 2e-16	***
hr21	29.4071	3.9362	7.471	8.74e-14	***
hr22	-8.5883	3.9332	-2.184	0.02902	*
hr23	-37.0194	3.9344	-9.409	< 2e-16	***
workingday	1.2696	1.7845	0.711	0.47681	
temp	157.2094	10.2612	15.321	< 2e-16	***
weathersitcloudy/misty	-12.8903	1.9643	-6.562	5.60e-11	***
weathersitlight rain/snow	-66.4944	2.9652	-22.425	< 2e-16	***
weathersitheavy rain/snow	-109.7446	76.6674	-1.431	0.15234	

Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 76.5 on 8605 degrees of freedom

Multiple R-squared: 0.6745, Adjusted R-squared: 0.6731

F-statistic: 457.3 on 39 and 8605 DF, p-value: < 2.2e-16

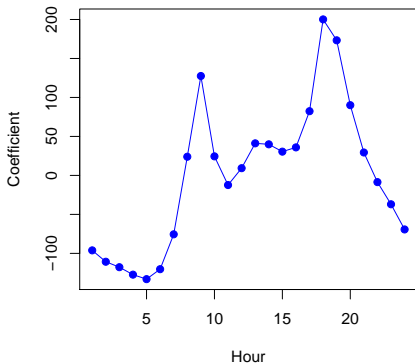
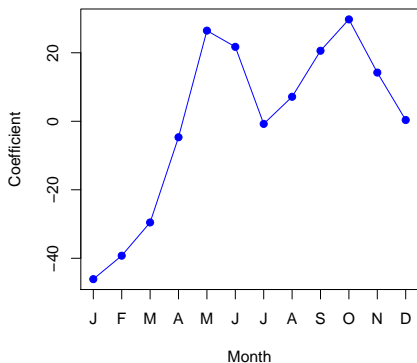
hr20	0.914022	0.003700	247.065	< 2e-16	***
hr21	0.616201	0.004191	147.045	< 2e-16	***
hr22	0.364181	0.004659	78.173	< 2e-16	***
hr23	0.117493	0.005225	22.488	< 2e-16	***
workingday	0.014665	0.001955	7.502	6.27e-14	***
temp	0.785292	0.011475	68.434	< 2e-16	***
weathersitcloudy/misty	-0.075231	0.002179	-34.528	< 2e-16	***
weathersitlight rain/snow	-0.575800	0.004058	-141.905	< 2e-16	***
weathersitheavy rain/snow	-0.926287	0.166782	-5.554	2.79e-08	***

Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

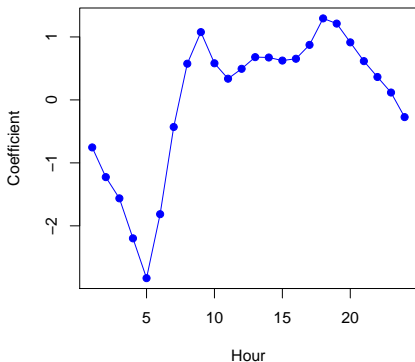
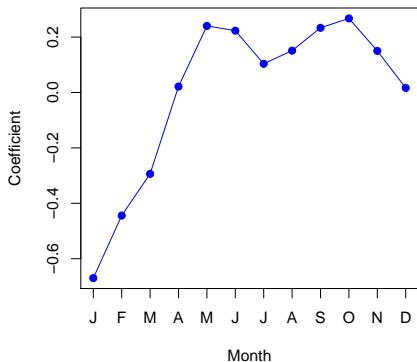
(Dispersion parameter for poisson family taken to be 1)

Null deviance: 1052921 on 8644 degrees of freedom
 Residual deviance: 228041 on 8605 degrees of freedom
 AIC: 281159

Number of Fisher Scoring iterations: 5



A least squares linear regression model was fit to predict bikers in the Bikeshare data set. Left: The coefficients associated with the month of the year. Bike usage is highest in the spring and fall, and lowest in the winter. Right: The coefficients associated with the hour of the day. Bike usage is highest during peak commute times, and lowest overnight [[James et al., 2021](#), Figure 4.13]




A Poisson regression model was fit to predict bikers in the Bikeshare data set. Left: The coefficients associated with the month of the year. Bike usage is highest in the spring and fall, and lowest in the winter. Right: The coefficients associated with the hour of the day. Bike usage is highest during peak commute times, and lowest overnight [James et al., 2021, Figure 4.15]


Important distinction: Poisson and linear regression models


- ① Coefficient associated with **workingday** is **statistically significant** under the Poisson regression model, but not under the linear regression model. **More realistic modeling!**
- ② **Mean-variance relationship:** in Poisson regression, we implicitly assume that **mean bike usage in a given hour equals the variance of bike usage during that hour.** while a constant variance in linear regression model.
- ③ **Nonnegative fitted values:** There are no negative predictions using the Poisson regression model.

- 1 Previous Episode: Discriminative Approaches for Classification Problems
 - Multiple logistic regression
 - Nearest-neighbour methods
 - High-dimensional data classification
- 2 Curse of Dimensionality
 - A synthetically generated Oil Flow data set
 - KNN learning algorithm
 - Curse of dimensionality for kNN
 - Lost in the immensity of high-dimensional spaces
 - The impact of high dimensionality on statistics is multiple!
- 3 Generalized Linear Models
 - Bikeshare data set: motivation example
 - Bikeshare data: linear regression
 - Bikeshare data: Poisson regression
 - Bikeshare data: linear regression and Poisson regression
 - Generalized linear models

Generalized linear models

 Recall that conditional on $X_{[P]} \equiv (X_1, \dots, X_P)$, Y belongs to a certain family of distributions: Gaussian or normal distribution for linear regression, Bernoulli distribution for logistic regression and Poisson distribution for Poisson regression. **What is the common thing?**


 $Y \in$ exponential family (including Gaussian, Bernoulli & Poisson).

 Each approach models the mean of Y as a function of the predictors.

① Linear regression $\mathbb{E}[Y|X_{[P]}] = \beta_0 + \sum_{p=1}^P \beta_p X_p.$

② Logistic regression $\mathbb{E}[Y|X_{[P]}] = \mathbb{P}(Y = 1|X_{[P]}) = \frac{e^{\beta_0 + \sum_{p=1}^P \beta_p X_p}}{1 + e^{\beta_0 + \sum_{p=1}^P \beta_p X_p}}.$

③ Poisson regression $\mathbb{E}[Y|X_{[P]}] = \lambda(X_{[P]}) = e^{\beta_0 + \sum_{p=1}^P \beta_p X_p}.$

 Using a link function η such that $\eta(\mathbb{E}[Y|X_{[P]}]) = \beta_0 + \sum_{p=1}^P \beta_p X_p.$

HOW. $\eta(\mu) = \mu$, $\eta(\mu) = \log(\mu/(1 - \mu))$, $\eta(\mu) = \log(\mu).$

References



Bishop, C. M. (2006).

Pattern Recognition and Machine Learning.

Information Science and Statistics. New York: Springer, 1 edition.

(Cited on pages 13, 14, 18, 20, 21, 22, 23, 25, 26, 27, and 28.)



Giraud, C. (2021).

Introduction to High-Dimensional Statistics.

Chapman & Hall/CRC Monographs on Statistics & Applied Probability. Taylor & Francis.

(Cited on pages 29 and 30.)



James, G., Witten, D., Hastie, T., & Tibshirani, R. (2021).

An Introduction to Statistical Learning: with Applications in R,
volume 2 of *Springer Texts in Statistics.*
Springer.

(Cited on pages 40, 41, 54, and 55.)