click Here

Bookmark this page

Q1

1.0/1.0 point (graded)

PCA is employed for

- (●) Dimensionality reduction
- ( ) Linear regression
- ( ) Dimensionality expansion
- ( ) Clustering

✔

Submit

Q2

1.0/1.0 point (graded)

Principal components of data can be found

- ( ) Via projection of data orthogonal to principal directions
- ( ) Same as the principal directions
- (●) Via projection of data along principal directions
- ( ) Orthogonal to the principal directions

✔

Submit

Q3

1.0/1.0 point (graded)

The data matrix $\mathbf{X}$ in the lecture has been defined as

- ( ) $\frac{1}{\sqrt{N-1}}\begin{bmatrix}\bar{\mathbf{x}}_1\\ \bar{\mathbf{x}}_2\\ \vdots\\ \bar{\mathbf{x}}_N\end{bmatrix}$
- (●) $\frac{1}{\sqrt{N-1}}\begin{bmatrix}\bar{\mathbf{x}}_1^T\\ \bar{\mathbf{x}}_2^T\\ \vdots\\ \bar{\mathbf{x}}_N^T\end{bmatrix}$
- ( ) $\frac{1}{\sqrt{N-1}}[\bar{\mathbf{x}}_1^T \quad \bar{\mathbf{x}}_2^T \quad \dots \quad \bar{\mathbf{x}}_N^T]$
- ( ) $\frac{1}{\sqrt{N-1}}[\bar{\mathbf{x}}_1 \quad \bar{\mathbf{x}}_2 \quad \dots \quad \bar{\mathbf{x}}_N]$

✔

Submit

Q4

1.0/1.0 point (graded)

The PCA routine can be imported in PYTHON as

- ○ from sklearn import PCA
- ◉ from sklearn.decomposition import PCA
- ○ import PCA
- ○ from decomposition import PCA

✔

Submit

Q5

1.0/1.0 point (graded)

Consider PCA routine called as
pca = PCA(n_components=2);
PCA can be applied and data X can be transformed in PYTHON as

- ○ Xp = pca.fit_transform(X)
- ○ Xp = pca.fittransform(X)
- ○ Xp = pca(X)
- ◉ Xp = pca.fit(X).transform(X)

✔

Submit

Q6

1.0/1.0 point (graded)

Consider the ML example below for prediction of sales based on advertising

Year	Sales (Million Euro)	Advertising (Million Euro)
1	651	23
2	762	26
3	856	30
4	1,063	34
5	1,190	43
6	1,298	48
7	1,421	52
8	1,440	57
9	1,518	58

In this example, Sales is the

- ○ Regressor
- ○ Regression coefficient
- ◉ Response

- ◯ Model error

✔

Submit

Q7

1.0/1.0 point (graded)

Consider the linear regression model below

$$y(k) = h_0 + h_1 x_1(k) + \cdots + h_n x_n(k) + \epsilon(k)$$

The quantities h_i are

- ◉ Regression coefficient
- ◯ Regressor
- ◯ Regressor
- ◯ Model error

✔

Submit

Q8

1.0/1.0 point (graded)

The linear regression module can be imported in PYTHON as

- ◯ from sklearn.linear_model import Regression
- ◯ from sklearn import LinearRegression
- ◉ from sklearn.linear_model import LinearRegression
- ◯ from sklearn import Regressio

✔

Submit

Q9

1.0/1.0 point (graded)

Metric used to characterize performance of linear regression is

- ◯ Only mean_squared_error but not r2_score
- ◯ Only r2_score but not mean_squared_error
- ◯ Neither r2_score nor mean_squared_error
- ◉ Both r2_score and mean_squared_error

✔

Submit

Q10

1.0/1.0 point (graded)

The linear regression model can be applied as

- reg = Regression()
  reg.fit(X_train, y_train)
- **reg = LinearRegression()**
  reg.fit(X_train, y_train)
- reg = LinearRegression()
  reg.fit(y_train, X_train)
- reg = Regression()
  reg.fit(y_train, X_train)

✔

Submit

Previous Next