Department of Mathematical and Computational Sciences
National Institute of Technology Karnataka, Surathkal

https://sam.nitk.ac.in/ sam@nitk.edu.in

MA722 - Foundations of Machine Learning Algorithms

Lab examination

1. In a manufacturing plant, the output quality score of a machine (denoted by `QualityScore`) is suspected to depend linearly on the machine temperature (`Temp`). Additional metadata such as operator group, time stamps, and offsets are recorded but are not part of the regression model.

   (a) Fit a simple linear regression model of the form:

   $$QualityScore = \beta_0 + \beta_1 \cdot Temp + \varepsilon.$$

   (b) Estimate β_0 and β_1 using ordinary least squares.

   (c) Interpret the slope parameter in the context of the manufacturing process.

   (d) Compute the residuals and the coefficient of determination (R^2).

   (e) Check whether the additional metadata fields (Group, Time, Offset) could be useful in an extended model.

Dataset

`QualityScore` (score)	`Temp` (°C)	`Noise` (aux)	`Group` (1–3)	`Time` (s)	`Offset` (numeric)	`OperatorID` (1–4)
78.2	62.5	0.4	1	0.5	0.0	2
80.1	64.0	0.2	1	1.0	0.0	1
82.7	66.2	0.5	1	1.5	0.1	2
70.3	55.1	0.3	2	0.4	0.0	3
72.4	56.8	0.2	2	0.8	0.1	4
68.9	53.7	0.5	3	1.2	0.0	1
85.2	68.9	0.3	3	0.3	0.0	3
88.1	70.4	0.4	2	1.7	0.1	4

2. (a) Generate polynomial features up to degree 3 for `EngineSize` vs `CO2Emissions`; fit and plot original points with fitted curve.

   (b) Train polynomial models degree 1–8 on noisy data; plot train vs test MSE and explain selection of degree.

   (c) Fit Ridge and Lasso on degree-6 polynomial features; compare coefficient shrinkage and test MSE.

Dataset

`CO2` (g/km)	`EngineSize` (L)	`Cylinders` (count)	`VehicleWeight` (kg)	`FuelType` (0=diesel,1=petrol)	`Horsepower` (hp)	`AerodynamicCd` (unitless)	`Mileage` (kmpl)
120	1.2	4	1050	1	90	0.31	18.5
180	2.0	4	1300	1	150	0.33	14.0
210	2.5	6	1600	0	200	0.36	11.2
95	1.0	3	950	1	70	0.30	20.1
250	3.5	6	1850	0	260	0.38	9.5
135	1.4	4	1100	1	100	0.32	17.0
170	1.8	4	1250	1	140	0.34	15.2
220	2.8	6	1700	0	220	0.37	10.8

3. (a) Implement logistic regression (sigmoid + cross-entropy) using gradient descent and plot the decision boundary for 2 selected features.
   (b) Fit with `sklearn`; compute accuracy, precision, recall, F1, plot ROC.
   (c) Train L2-regularized logistic regression; vary regularization strength and plot coefficient magnitudes.

Dataset

Admit (0/1)	GRE (out of 340)	GPA (4.0)	Rank (1-4)	ResearchExp (0/1)	CourseworkScore (0-100)	RecommendationStrength (1-5)
1	330	3.8	1	1	92	5
0	300	3.0	3	0	78	3
1	315	3.6	2	1	85	4
0	280	2.8	4	0	70	2
1	340	3.9	1	1	95	5
0	290	3.1	3	0	80	3
1	320	3.7	2	1	88	4
0	275	2.7	4	0	68	2

4. (a) Fit a linear SVM to separable 2-class data and plot decision boundary and margins.
   (b) Train SVM with polynomial and RBF kernels; compare test accuracies and decision regions.
   (c) Grid-search over `C` and `gamma`; visualize validation heatmap.

Dataset

Label (0/1)	F1	F2	F3	F4	F5	F6	F7
0	-1.2	0.5	0.8	-0.4	1.0	0.2	-0.9
1	1.1	0.9	-0.2	1.3	-0.1	0.5	1.0
0	-0.8	0.3	0.6	-0.2	0.7	0.1	-0.6
1	1.5	1.2	-0.5	1.6	0.0	0.7	1.3
0	-1.0	0.4	0.7	-0.3	0.9	0.15	-0.8
1	0.9	0.8	-0.1	1.1	-0.05	0.45	0.95
0	-1.3	0.6	0.9	-0.5	1.05	0.25	-0.95
1	1.2	1.0	-0.3	1.4	0.02	0.6	1.1

5. (a) Implement MLE for mean and variance for a Gaussian sample and compare to NumPy estimates.
   (b) Estimate p for Bernoulli data; plot the log-likelihood as a function of p.
   (c) Estimate λ from counts (Poisson) and numerically verify concavity of log-likelihood.

Dataset

ObsValue	Type (G=Gaussian, P=Poisson)	Count (counts)	BernSample (0/1)	Group (1-3)	Time (s)	Offset (numeric)
12.4	G			1	0.5	0.0
11.9	G			1	1.0	0.0
13.1	G			1	1.5	0.0
5	P	5		2	0.2	0.0
2	P	2		2	0.3	0.0
7	P	7		3	0.4	0.0
1			0	2	0.6	0.0
0			1	3	0.7	0.0

6. (a) Compute covariance matrix, eigenvalues/eigenvectors, and project onto top 2 PCs.

(b) Use `sklearn` PCA; plot explained variance ratio and scatter PC1 vs PC2.

(c) Reduce to k components and compute reconstruction error for $k = 1, \ldots, 5$.

Dataset

S1	S2	S3	S4	S5	S6	S7	Label (optional)
2.5	1.2	0.5	3.1	1.0	0.8	2.0	A
3.0	1.5	0.8	2.9	1.2	0.7	2.3	A
1.8	0.9	0.4	3.4	0.8	0.9	1.7	B
2.2	1.1	0.6	3.0	1.1	0.85	1.9	A
3.5	1.8	1.0	2.6	1.5	0.6	2.7	A
1.6	0.7	0.3	3.6	0.7	1.0	1.5	B
2.9	1.4	0.7	2.8	1.3	0.65	2.2	A
2.0	1.0	0.5	3.2	0.95	0.9	1.8	B

7. (a) For a small dataset compute Gini impurity for candidate splits and verify with `sklearn`.

   (b) Train trees with varying `max_depth`; plot depth vs accuracy and pick optimal depth.

   (c) Fit a decision tree regressor and visualize using `plot_tree` or export_graphviz.

Dataset

Target (0/1 or numeric)	X1	X2	X3	X4	Cat (A/B/C)	Score (0-100)
1	2.1	0.5	1.2	3.4	A	78
0	1.8	0.3	0.9	2.8	B	65
1	2.5	0.7	1.4	3.8	A	82
0	1.2	0.2	0.4	2.1	C	50
1	2.8	0.9	1.6	4.0	A	88
0	1.0	0.1	0.2	1.8	C	45
1	2.3	0.6	1.3	3.6	B	80
0	1.5	0.25	0.6	2.3	B	60

8. (a) Derive MAP estimator for Gaussian mean with Gaussian prior; implement and compare to MLE on synthetic data.

   (b) Assume Gaussian prior on weights; compute posterior mean closed-form and use it for prediction.

   (c) Plot MLE vs MAP estimates as a function of sample size and discuss shrinkage effect.

Dataset

Obs	Feature1	Feature2	Feature3	Outcome	PriorMean	PriorVar
12.2	1.1	0.5	2.0	11.5	10.0	4.0
13.0	1.3	0.6	2.2	12.7	10.0	4.0
11.5	0.9	0.4	1.8	10.8	10.0	4.0
14.1	1.5	0.7	2.6	13.4	10.0	4.0
12.8	1.2	0.55	2.1	12.0	10.0	4.0
13.6	1.4	0.65	2.3	12.9	10.0	4.0
11.9	1.0	0.45	1.9	11.2	10.0	4.0
12.5	1.15	0.52	2.05	11.9	10.0	4.0

9. (a) Construct a simple 4-node Bayesian network (e.g., Weather, Sprinkler, WetGrass, Rain) using `pgmpy` or similar and visualize.

   (b) Perform variable elimination to compute $P(Rain \mid WetGrass = true)$.

   (c) Learn CPDs from synthetic data and compare learned CPDs to ground truth.

Dataset

Weather (Sunny/Cloudy/Rainy)	Sprinkler (0/1)	Rain (0/1)	WetGrass (0/1)	Humidity (0-100)	Temp (C)	TimeOfDay (Morning/Afternoon/Evening)
Sunny	1	0	1	40	28	Morning
Cloudy	0	1	1	75	22	Afternoon
Rainy	0	1	1	90	18	Evening
Sunny	1	0	0	35	30	Morning
Cloudy	0	0	0	70	24	Afternoon
Rainy	0	1	1	88	19	Evening
Sunny	1	0	1	42	27	Morning
Cloudy	0	1	1	80	21	Afternoon

10. (a) Implement K-means from scratch and visualize cluster centers and assignments per iteration.
    (b) Run K-means for $k = 1, \ldots, 10$ and plot inertia vs k to choose k.
    (c) Compute silhouette scores for different k and plot silhouette coefficients.

Dataset

X	Y	F1	F2	F3	F4	F5	F6
1.0	2.1	0.5	1.2	3.0	0.8	2.2	1.0
1.3	1.9	0.6	1.1	2.8	0.7	2.0	0.9
5.0	4.8	3.1	2.9	0.5	3.2	0.4	2.7
4.8	5.2	3.0	3.1	0.6	3.0	0.6	2.9
8.0	1.0	1.8	0.5	4.1	1.5	3.5	0.8
7.5	1.3	1.7	0.4	4.0	1.6	3.2	0.9
5.2	5.0	3.2	3.0	0.45	3.1	0.5	2.8
1.1	2.0	0.55	1.15	3.05	0.82	2.15	0.98

*****