Machine learning (R22CS602PC)

R22

III Year II Sem Supply Exam - Dec - 2025 (19/12/25)

Key Question Paper.

PART-A

1) a) Interpret handwriting recognition as well Posed learning Problem.

Sol: Handwriting recognition is a well Posed learning Problem because the task is to identify handwritten characters, the experience consists of labeled handwriting samples, and Performance is measured by recognition accuracy.

2) b) Consider the hypothesis $h_1 = \langle Sunny, ?, ?, ?, ?, Change \rangle$. Justify the Consistency of h_1 over the instance (or example) $d_1 = \langle Sunny, warm, Normal, strong, warm, same \rangle$ whose target label is "Yes."

Ans: The hypothesis $h_1 = \langle Sunny, ?, ?, ?, ? \rangle$ is consistent because the wildcard ? can match any attribute value, so is correctly classifies all Possible Positive training instances as "Yes".

3) c) Differenciate between a Perceptron and multilayer Perceptron,

Ans: A Perceptron is a single layer neural network that solves only linear Problems, whereas multilayer Perceptron has hidden layers and can solve non-linear Problems.

4) d) State SVM Principles.

Ans: Support vector machine works on the Principle of finding an optimal hyperplane that maximizes the margin between different classes.

e) Define Entropy.

Ans: Entropy is a measure of uncertainty or impurity in a dataset commonly used in decision trees to determine the best attribute for splitting the data.

f) List any two algorithms which use bagging and boosting.

Ans: Bagging : Random Forest, Bagged decision trees
Boosting : AdaBoost, Gradient Boosting.

g) Perform single-Point Crossover at Position 3 between binary strings : A = 110101 , B = 001110

Ans: Single-Point Crossover at Position 3

A = 110|101
B = 001|110

Off Spring 1 : 110110
Off Spring 2 : 001101

h) LDA is best suited for which type of problem : classification or Regression.

Ans: Linear Discriminant Analysis (LDA) is best suited for classification Problems.

i) Write state update rule in Q-Learning?

Ans: $Q(s,a) \leftarrow Q(s,a) + \alpha[r + \gamma \max Q(s',a') - Q(s,a)]$

Where α = learning rate
γ = discount factor s' = next state.
r = reward

1) Differentiate between Sampling with and without Replacement.

Ans: With Replacement: An item can be selected more than once in the sample.

Without Replacement: An item cannot be selected again once it is chosen.

PART-B

2) Describe the Linear Regression algorithm, Derive the formula for the weights of a linear regression model.

Ans: Linear Regression Algorithm and Derivation of Weight Formula

(i) Introduction to Linear Regression.

Linear Regression is a Supervised learning algorithm used to model the relationship between a dependent variable (output) and one or more independent variables (inputs) by fitting a linear equation to observed data.
It is mainly used for Prediction and regression Problems.

(ii) Linear Regression Model:

For a dataset with n samples and m features;

$$y = w_0 + w_1x_1 + w_2x_2 + \cdots + w_mx_m$$

Where: y = Predicted output, w_0 = bias

$x_1, x_2 \cdots x_m$ = i/p features. $w_1, w_2 \cdots w_m$ = weights (Parameters)

(iii) Objective of Linear Regression:

The goal is to find weights W such that the difference between Predicted and actual values is minimized, This difference is measured using the Mean Squared Error (MSE) cost function:

$$J(w) = \frac{1}{2n} \sum_{i=1}^{n} (y_i - \hat{y}_i)^2$$

(iv) Error Function

Using matrix function

$$J(w) = \frac{1}{2} (y - Xw)^T (y - Xw)$$

(v) Derivation of weight Formula (Normal equation)

Take derivative

Step 1 : $\frac{\partial J}{\partial w} = -X^T (y - Xw)$

Step 2 : Set derivative to zero

$$X^T (y - Xw) = 0$$

Step 3 : Expand

$$X^T y = X^T X w$$

Step 4 : Solve for W

$$w = (X^T X)^{-1} X^T y$$

(vi) Final weight Formula : $W = (X^T X)^{-1} X^T y$.

(vii) Algorithmic steps for Linear Regression:

① Collect training dataset (X, y) ② Assume linear model ($y = Xw$) ③ Define cost function (MSE) ④ Minimize cost function ⑤ Compute weight using normal equation ⑥ Use the model for Prediction

3) Find the Version Space for the given data using Candidate Elimination algorithm.

Origin	manufacturer	Color	Decade	Type	Example Type
Japan	Honda	Blue	1980	Economy	Positive
Japan	Toyota	Green	1970	Sports	Negative
Japan	Toyota	Blue	1990	Economy	Positive
USA	Chrysler	Red	1980	Economy	Negative
Japan	Honda	White	1980	Economy	Positive

Ans: Using the Candidate Elimination Algorithm, we find the Version Space by computing the Specific boundary (S) and general boundary (G).

Given attributes.

< Origin, Manufacturer, Color, Decade, Type >

Training examples (from the table)

1. <Japan, Honda, Blue, 1980, Economy> → Positive
2. <Japan, Toyota, Green, 1970, Sports> → Negative.
3. <Japan, Toyota, Blue, 1990, Economy> → Positive
4. <USA, Chrysler, Red, 1980, Economy> → Negative
5. <Japan, Honda, White, 1980, Economy> → Positive.

Step 1. Initialize.

$S_0 = (\emptyset, \emptyset, \emptyset, \emptyset, \emptyset)$

$G_0 = (?, ?, ?, ?, ?)$

Step 2: Process Positive examples

After processing all positive examples, the specific boundary becomes

$$S = \langle \text{Japan}, ?, ?, 1980, \text{Economy} \rangle$$

Step 3: Process Negative examples.

After removing hypotheses in G that cover negative examples and keeping only those consistent with S, the general boundary is

$$G = \{ \langle \text{Japan}, ?, ?, ?, \text{Economy} \rangle, \langle ?, ?, ?, 1980, \text{Economy} \rangle \}$$

The version space is the set of all hypothesis between S and G

Where:

- Specific boundary (S): $\langle \text{Japan}, ?, ?, 1980, \text{Economy} \rangle$
- General boundary (G):

$$\{ \langle \text{Japan}, ?, ?, ?, \text{Economy} \rangle, \langle ?, ?, ?, 1980, \text{Economy} \rangle \}$$

This represents all hypotheses consisted with the given training data.

4) Derive back Propagation rule for both hidden and o/p layers.

Ans: Consider a simple feedforward neural network:

→ Input : x
→ Hidden layer $h = f(z^h) = f(W^h x + b^h)$
→ Output layer $y_{Pred} = g(z^o) = g(W^o h + b^o)$
→ Loss function : $L(y_{Pred}, y)$

Here: W^h, b^h = Weights and biases of hidden layer.
W^o, b^o = Weights and biases of output layer.
f = activation function in hidden layer
g = activation function in output layer.

We aim to compute the gradients:

$$\frac{\partial L}{\partial W^o}, \frac{\partial L}{\partial b^o}, \frac{\partial L}{\partial W^h}, \frac{\partial L}{\partial b^h}$$

(i) Gradients for o/p layer.

For o/p layer, using chain rule:

$$\frac{\partial L}{\partial W^o} = \frac{\partial L}{\partial z^o} \cdot \frac{\partial z^o}{\partial W^o}$$

but $z^o = W^o h + b^o \Rightarrow \frac{\partial z^o}{\partial W^o} = h^T$

Define the Error term at o/p layer as: $\delta^o = \frac{\partial L}{\partial z^o} = \frac{\partial L}{\partial y_{Pred}} \odot g'(z^o)$

Then $\frac{\partial L}{\partial W^o} = \delta^o (h)^T \Rightarrow \frac{\partial L}{\partial b^o} = \delta^o$

(ii) Gradients for Hidden Layer

For the hidden layer, the error Propagates backward from the output:

$$\frac{\partial L}{\partial W^h} = \frac{\partial L}{\partial z^h} \cdot \frac{\partial z^h}{\partial W^h}$$

$$z^h = W^h x + b^h \Rightarrow \frac{\partial z^h}{\partial W^h} = x^T$$

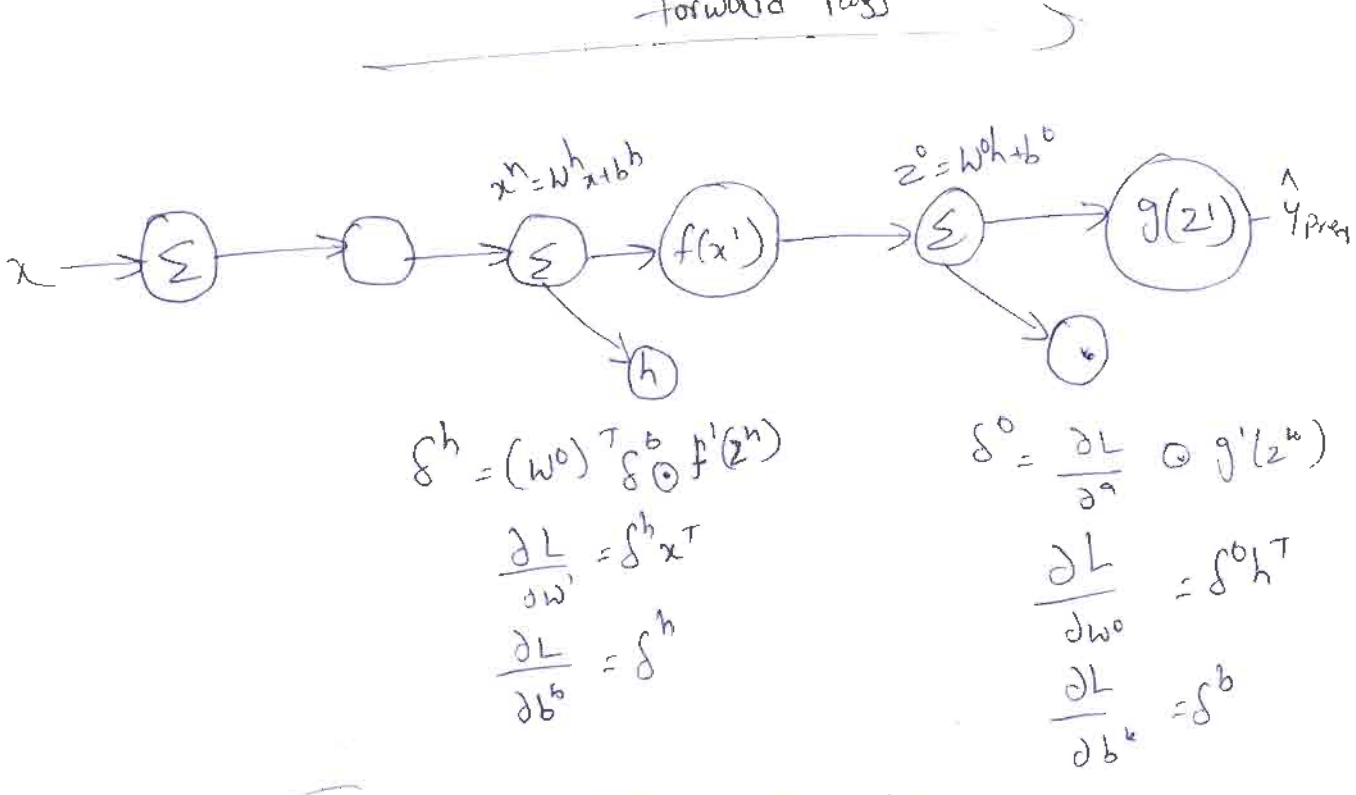
Error term in hidden layer

$$\delta^h = \frac{\partial L}{\partial z^h} = (W^o)^T \delta^o \odot f'(z^h)$$

So the gradient updates are:

$$\frac{\partial L}{\partial W^h} = \delta^h (x)^T$$

$$\frac{\partial L}{\partial b^h} = \delta^h$$



5) Explain the role of basis functions in interpolation. Describe how splines are used as basis functions for smooth interpolation of data points.

Ans: Interpolation is the process of estimating values of a function $f(x)$ at points where it is not explicitly known, using a set of known data points (x_i, y_i).

A basis function is a building block used to construct the interpolating function. mathmatically, the interpolated function is expressed as: $\boxed{f(x) = \sum_{i=1}^{n} c_i \phi_i(x)}$ where $\phi_i(x)$ = basis functions

c_i = coefficients determined by the known data points

n = No. of data points.

Role of basis functions:

(i) They define the shape of the interpolating function b/w points.

(ii) They allow interpolation to be written as a linear combination, making computations systematic.

(iii) They determine the smoothness and continuity of the interpolated function.

(iv) Different choices of basis functions lead to different interpolation methods (e.g., Polynomials, Piecewise linear, splines, Radial basis functions).

example: In Polynomial interpolation, the basis functions are $\phi_i(x) = x^{i-1}$.

6) Explain the Concepts of bagging and boosting in ensemble learning. Compare their mechanisms and use cases.

Ans: Ensemble learning Combines multiple model to improve overall Performance Compared to a single model. The main idea is:

Final Prediction - Aggregation of Predictions from multiple models,

Two Popular ensemble techniques are Bagging and Boosting.

→ Bagging (Bootstrap aggregating):

Bagging is designed to reduce Variance and avoid overfitting by training models on different Subsets of data.

Example algorithm for Bagging is Random forest algorithm.

→ Boosting

Boosting is designed to reduce bias by Sequentially training models, each focusing on the mistakes of the Previous ones.

Example algorithms are Adaboost, Gradient Boosting machine, XG Boost, LightGBM, CatBoost.

So finally choose Bagging, when models overfit (high Variance) boosting when models underfit (high bias).

7) Construct decision tree for the following data.

Id	outlook	Temparature	Humidity	wind	Play Tennis.
1.	Sunny	Hot	High	weak	0
2.	Sunny	Hot	High	strong	0
3.	Overcast	Hot	High	weak	1
4.	Rainy	Mild	High	weak	1
5.	Rainy	Cool	Normal	Weak	1
6.	Rainy	Cool	Normal	strong	0
7.	Overcast	Cool	Normal	strong	1
8.	Sunny	Mild	High	wear	0

Ans: step1: Identify the Target and Attributes.

→ Target : Play Tennis (Yes/No)

→ Attributes : Outlook, Temperature, Humidity, Wind

step2: Calculate the Entropy of the Entire Dataset.

$$\text{Entropy}(S) = -P_+ \log_2 P_+ - P_- \log_2 P_-$$

P_+ = Proportion of Yes (Play tennis = 1) = 5/8

P_- = Proportion of No (Play tennis = 0) = 3/8

$$\text{Entropy}(S) = -\frac{5}{8}\log_2\frac{5}{8} - \frac{3}{8}\log_2\frac{3}{8} \approx 0.954$$

Step3: Compute Entropy for each Attribute.

a) Outlook

Outlook	Play=Yes	Play=No	Entropy
Sunny	2	1	$E = -2/3 \log_2 2/3 - 1/3 \log_2 1/3 \approx 0.918$
Overcast	2	0	$E = 0$
Rainy	1	2	$E = 0.918$

$$\text{Info}(S, \text{Outlook}) = \frac{3}{8}(0.918) + \frac{2}{8}(0) + \frac{3}{8}(0.918) \approx 0.689$$

$$\text{Gain}(S, \text{Outlook}) = 0.954 - 0.689 = 0.265$$

b) Temperature.

Temperature	Play=Yes	Play=No	Entropy
Hot	2	2	$E = 1$
Mild	2	0	$E = 0$
cool	1	1	$E = 1$

$$\text{Info}(S, \text{Temp}) = \frac{4}{8}(1) + \frac{2}{8}(0) + \frac{2}{8}(1) = 1$$

$$\text{Gain}(S, \text{Temp}) = 0.954 - 1 = -0.046 \approx 0$$

c) Humidity

Humidity	Play(Yes)	Play(No)	Entropy
High	3	2	$E = 0.971$
Normal	2	1	$E = 0.918$

$$\text{Info}(S, \text{Humidity}) = \frac{5}{8}(0.971) + \frac{3}{8}(0.918) \approx 0.954$$

$$\text{Gain}(S, \text{Humidity}) = 0.954 - 0.954 \approx 0$$

d) wind

wind	Play(Yes)	Play(No)	Entropy
Weak	4	1	E=0.722
Strong	1	2	E=0.918

$$Info(S, wind) = \frac{5}{8}(0.722) + \frac{3}{8}(0.918) \approx 0.788$$

$$Gain(S, wind) = 0.954 - 0.788 = 0.166.$$

∴ Outlook has highest gain = 0.265 ⟶ Root node.

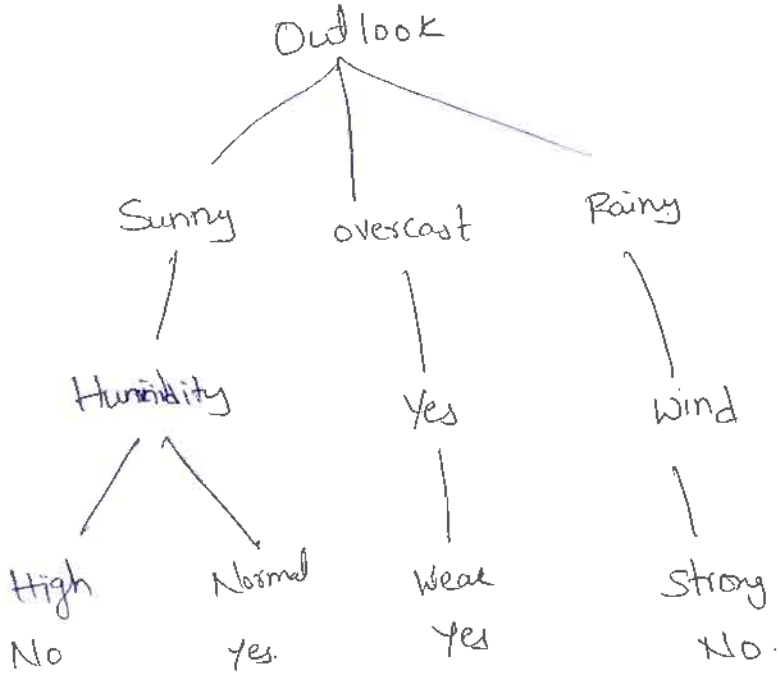


Fig: Desicion Tree.

8) Describe the LDA (linear discriminant analysis) algorithm. How does LDA Preserve local neighbourhood information while reducing dimensions.

Ans: LDA is a supervised dimensionality reduction technique used to project high dimensional data onto a lower-dimensional space while maximizing class separability.

Algorithm steps:

1. Compute the mean vectors:
   For each class i, calculate the mean vector M_i.

2. Compute scatter matrices:
   → within-class scatter matrix (S_w): Measures the spread of samples within each class:

$$S_w = \sum_{i=1}^{C} \sum_{x \in C_i} (x - M_i)(x - M_i)^T$$

   → Between-class scatter matrix (S_b): Measures the spread b/w class means

$$S_b = \sum_{i=1}^{C} n_i (M_i - M)(M_i - M)^T$$

   where M = overall mean, n_i = no. of samples in class i.

3. Compute the Projection Matrix:
   Find the matrix W that maximizes the ratio of b/w-class to within-class scatter:

$$W = \arg\max |W^T S_b W| / |W^T S_w W|$$

   generalized eigenvalue - $S_b W = \lambda S_w W$

4. Project data:
   Transform the original data x into lower-dimensional space: $Y = W^T x$.

9. Explain the concept of PCA (Principal Component Analysis) to reduce the dataset from 2D to 1D.

Ans: → PCA is a dimensionality reduction technique used in data analysis. It is unsupervised learning algorithm.

→ It transforms the original features into a new set of uncorrelated variable called Principal components.

→ The goal is to capture as much variance as possible in fewer dimensions.

→ Suppose team has a dataset with two features: x_1 and x_2.

→ Sometimes, these two features are correlated, so a lot of the information is redundant.

→ PCA finds a new axis (Principal component) that best represents the data in a single dimension.

Steps for Process 2D → 1D PCA

Step1: Center the data.

→ subtract the mean of each feature to center the data at the origin $X_{centered} = X - mean(X)$

step2: Compute the covariance matrix

→ Coveriance measures how features vary together. For 2D

$$Cov(x) = \begin{bmatrix} Var(x_1) & Cov(x_1, x_2) \\ Cov(x_1, x_2) & Var(x_2) \end{bmatrix}$$

Step3: Compute eigenvectors and eigenValues.

Step4: Project the data onto first Principal component.

steps: Result.

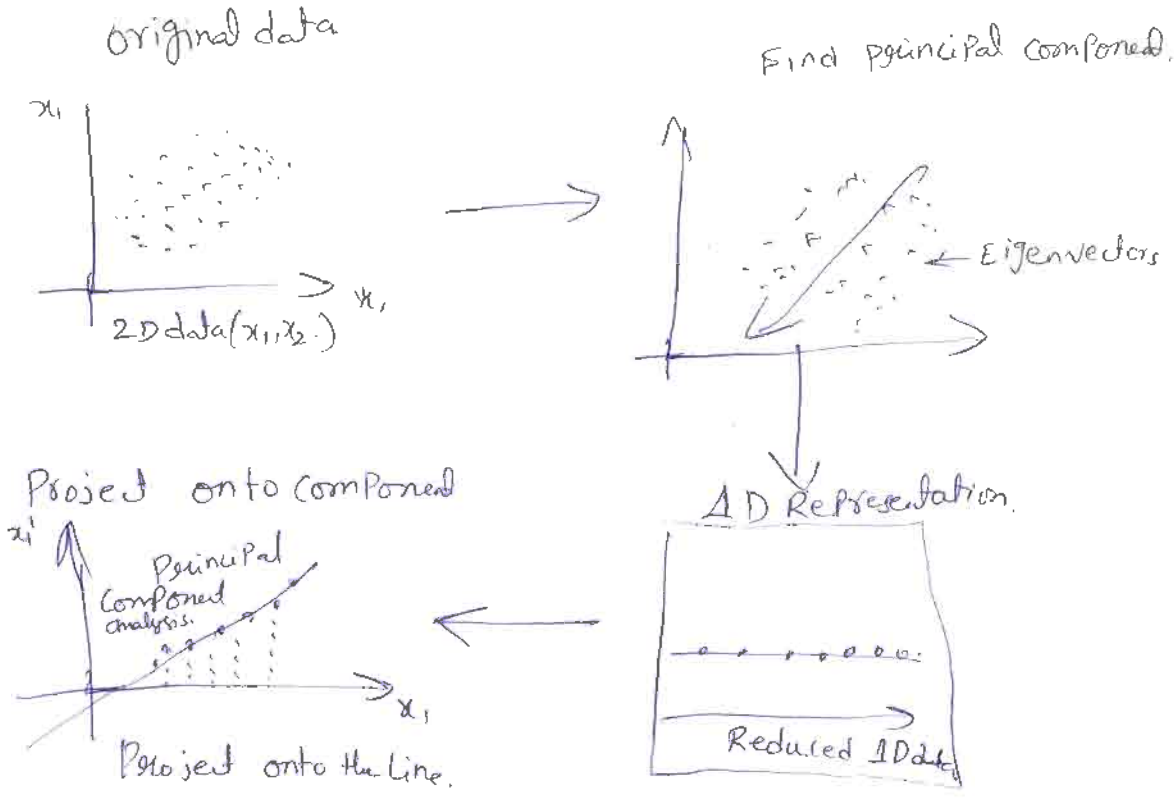


Fig: PCA: Reducing 2D Data to 1D

10) Compare and contrast Reinforcement learning with Supervised and unsupervised Learning,

Ans: Definition.

Learning Type	Definition
Supervised Learning (SL)	Learns a mapping from I/P to o/p using labeled data.
Unsupervised learning (UL)	Finds Patterns or structures in unlabeled data without explicit outputs.
Reinforcement Learning (RL)	Learns by interacting with an environment, taking actions to maximize cumulative reward.

Data Requirment :

Learning type	Data Requirement
SL	Requires leabled dataset (features + target labels)
UL	Works with unlabeled dataset (only features)
RL	Needs an environment and feedback (reward / Punishment) for actions, not explicit labels.

Feed back type :

Learning type	Feedback.
SL	Immediate and exact
UL	No explicit feedback
RL	Delayed feedback

Example Applications

Learniy type	Example
SL	Image classification, regression, Spam detection
UL	Customer segmentation, anomaly detection, PCA.
RL	Game AI (Chess, Go), autonomous driving.

11) Describe the importance of the Proposal distribution in MCMC (Markov chain monte Carlo) algorithms,

Ans: In MCMC algorithms, the Proposal distribution plays a central role in deterring how efficiently and accurately the target probability distribution is sampled.

→ The Proposal distribution, usually denoted as $q(x'/x)$, is the rule used to generate a candidate (Proposed) Sample x' from the current state x.

→ MCMC algorithms such as metropolis-Hastings use this Proposed sample and then decide whether to accept or reject it.

Affects Acceptance Probability.

→ In metropolis-Hastings, the acceptance Probability depends on both the target distribution and the Proposal distribution.

$$\alpha = min\left(1, \frac{\pi(x')\, q(x/x')}{\pi(x)\, q(x'/x)}\right)$$

A well chosen Proposal increases the chances of accepting useful moves, improving efficiency