DATA MINING LAB  
**You said:**

1 List all the categorical (or nominal) attributes and the real-valued attributes seperately. Attributes:- 1. checking\_status 2. credit history 3. purpose 4. savings\_status 5. employment 6. personal status 7. debtors 8. property 9. installment plans 10. housing 11. job 12. telephone 13. foreign worker 14. duration 15. credit amount 16. credit amount 17. residence 18. age 19. existing credits 20. num\_dependents 2 What attributes do you think might be crucial in making the credit assessment ? Come up with some simple rules in plain English using your selected attributes. 3 3 Train a Decision Tree using the complete dataset asthe training data. Report the model obtained after training. 3 4 Suppose you use your above model trained on the complete dataset, and classify credit good/bad for each of the examples in the dataset. What % of examples can you classify correctly? Why do you think you cannot get 100 % training accuracy? 3 5 Solving the problem encountered in the previous question is using cross-validation? Describe what cross-validation is briefly. Train a Decision Tree again using crossvalidation and report your results. Does your accuracy increase/decrease? Why? 3 6 Another question might be, do you really need to input so many attributes to get good results? Maybe only a few would do. For example, you could try just having attributes 2, 3, 5, 7, 10, 17 (and 21, the class attribute (naturally)). Try out some combinations. (You had removed two attributes in problem 7. Remember to reload the arff data file to get all the attributes initially before you start selecting the ones you want.) 3 7 Train your Decision Tree again and report the Decision Tree and cross-validation results. Are they significantly different from results? 3 8 Do you think it is a good idea to prefer simple decision trees instead of having long complex decision trees? How does the complexity of a Decision Tree relate to the bias of the model? 3 9 How make Decision Trees simpler by pruning the nodes. One approach isto use Reduced Error Pruning - Explain this idea briefly. Try reduced error pruning for training your Decision Trees using cross-validation and report the Decision Tree you obtain ? Also, report your accuracy using the pruned model. Does your accuracy increase? 3 10 How can you convert a Decision Trees into "if-then-else rules". Make up your own small Decision Tree consisting of 2-3 levels and convert it into a set of rules. There also exist different classifiers that output the model in the form of rules - one such classifier in Weka is rules.PART, train this model and report the set of rules obtained. Sometimes just one attribute can be good enough in making the decision, yes, just one ! Can you predict what attribute that might be in this dataset? OneR classifier uses a single attribute to make decisions (it chooses the attribute based on minimum error). Report the rule obtained by training a one R classifier. Rank the performance of j48, PART and oneR. 3 11 Implement K-means clustering and Justify your answers. 3 12 Implement SVM algorithm and report accuracy and compare the result with decision tree results. all possibel viva q and a pls

**ChatGPT said:**

Updated saved memory

Here are **possible viva questions and their answers** based on your credit assessment project using Decision Trees, K-Means Clustering, and SVM:

**🔹 1. What are categorical and real-valued attributes in your dataset?**

**Categorical (Nominal):**

* checking\_status
* credit history
* purpose
* savings\_status
* employment
* personal status
* debtors
* property
* installment plans
* housing
* job
* telephone
* foreign worker

**Real-Valued:**

* duration
* credit amount
* residence
* age
* existing credits
* num\_dependents

**🔹 2. Which attributes are crucial for credit assessment? Any sample rules?**

**Important Attributes:**

* credit history
* checking\_status
* credit amount
* employment
* existing credits
* purpose

**Sample Rules in Plain English:**

* If credit history is ‘no credits/all paid’ and checking\_status is ‘>200’, then credit is good.
* If employment is ‘<1 year’ and credit amount is > 5000, then credit is likely bad.
* If savings\_status is ‘<100’ and existing credits > 2, then credit may be bad.

**🔹 3. What model did you get after training the Decision Tree on the complete dataset?**

* Model is generated using the J48 classifier (Weka's implementation of C4.5).
* Example:

makefile

CopyEdit

credit\_history = no\_credits → good

credit\_history = critical/other existing credit → bad

checking\_status = <0 → bad

else → good

* The tree uses splits on features like credit history, checking\_status, and credit amount.

**🔹 4. What % accuracy do you get when training and testing on the same dataset?**

* You may get **90-100% accuracy**, but this is misleading.
* Reason: **Overfitting** – the model memorizes training data and fails on unseen data.

**🔹 5. What is cross-validation? Why is it important?**

* **Cross-validation** is a resampling technique where the dataset is divided into k folds.
* Each fold is used as a test set once while the rest are used for training.
* Helps get a realistic estimate of model performance.
* Accuracy may drop slightly but will reflect generalization better.

**🔹 6. Can you use fewer attributes? Try using attributes 2, 3, 5, 7, 10, 17, and 21.**

* Yes, models can be trained with fewer attributes.
* These attributes still capture essential behavioral and financial patterns.
* You can compare model performance with full and reduced attribute sets.

**🔹 7. What happens when you retrain the Decision Tree with selected attributes?**

* The Decision Tree becomes **simpler and more interpretable**.
* Accuracy might reduce slightly, but the tradeoff is better generalization and faster inference.
* If accuracy remains comparable, it's better to use the simpler model.

**🔹 8. Is it better to prefer simple trees over complex ones?**

* Yes. Simple trees are:
  + More interpretable
  + Less likely to overfit
* Complex trees have **low bias but high variance**, meaning they fit training data too closely and fail on new data.

**🔹 9. What is Reduced Error Pruning? Does it help?**

* It’s a pruning method that removes nodes if accuracy does not drop on a validation set.
* Simplifies the tree without reducing predictive performance.
* Usually leads to better generalization and may increase accuracy on test data.

**🔹 10. How do you convert a decision tree into if-then-else rules?**

**Example Tree:**

java

CopyEdit

if credit\_history = good:

if credit\_amount < 2000 → credit = good

else → credit = bad

else → credit = bad

**Rules:**

* IF credit\_history = good AND credit\_amount < 2000 THEN credit = good
* ELSE credit = bad

**PART Classifier Rules (Sample):**

* IF checking\_status = <0 AND duration > 12 THEN credit = bad
* IF employment = >7 yrs THEN credit = good

**OneR Rule (Example):**

* IF credit\_history = ‘critical/other existing credit’ → credit = bad
* ELSE → credit = good

**Ranking based on performance (may vary):**

1. J48 (Decision Tree) – Best tradeoff of interpretability and accuracy
2. PART – Slightly simpler but effective
3. OneR – Fast and interpretable but lower accuracy

**🔹 11. How did you implement K-Means? Justify.**

* K-Means was used to cluster customers based on features like age, credit amount, duration.
* Helps in **unsupervised grouping** to find risk clusters.
* Chose k using **elbow method**.
* Clusters help in pre-segmenting credit profiles.

**🔹 12. How did SVM perform compared to Decision Trees?**

* SVM usually provides **higher accuracy** with proper kernel selection.
* Works well on high-dimensional spaces.
* Harder to interpret compared to Decision Trees.
* If non-linear data, RBF or polynomial kernels outperform linear models.

**🔹 Other Possible Viva Questions:**

**Q: Why use Decision Trees for credit scoring?**  
A: They are interpretable, can handle both categorical and numerical data, and provide quick rule-based outputs.

**Q: What's the main drawback of Decision Trees?**  
A: Overfitting, especially with noisy or imbalanced data.

**Q: What’s the difference between J48 and CART?**  
A: J48 is Weka’s version of C4.5 (uses information gain ratio), while CART uses Gini index and binary splits.

**Q: How is the performance measured?**  
A: Using accuracy, precision, recall, F1-score, and cross-validation.

**Q: What is entropy and information gain?**  
A: Entropy measures impurity. Information gain is the reduction in entropy after a split.

Would you like this as a **PDF cheat sheet**, **Canva slide-ready version**, or **interactive quiz format**?

**You said:**

this too pls

**ChatGPT said:**

Based on your **syllabus image**, here are **viva questions and answers** organized **unit-wise** relevant to your Credit Assessment project and the topics mentioned:

**✅ UNIT 1: Introduction to Data Mining**

**🔹 1. What is Data Mining?**

**Ans:** Data Mining is the process of discovering patterns and knowledge from large amounts of data using techniques from statistics, machine learning, and databases.

**🔹 2. What is KDD?**

**Ans:** Knowledge Discovery in Databases (KDD) is the overall process of converting raw data into useful information. Data mining is one step in this process.

**🔹 3. What is the role of data preprocessing in Data Mining?**

**Ans:** Data preprocessing involves:

* Cleaning (handling missing values/noise)
* Integration (merging data sources)
* Reduction (reducing volume)
* Transformation (scaling, normalization)
* Discretization and Binarization (converting continuous data)

**🔹 4. What are similarity and dissimilarity measures?**

**Ans:** These are quantitative ways to measure how alike or different two data objects are. Example: Euclidean distance, Cosine similarity.

**✅ UNIT 2: Association Rules**

**🔹 5. What is the APRIORI algorithm used for?**

**Ans:** APRIORI is used to generate frequent item sets and association rules in market basket analysis.

**🔹 6. Define support and confidence.**

* **Support:** Frequency of itemset in the dataset.
* **Confidence:** Likelihood of item Y being bought if item X is bought.

**🔹 7. Difference between FP-Growth and APRIORI?**

**Ans:**

* APRIORI generates candidate sets and is slower.
* FP-Growth uses a prefix tree and avoids candidate generation → faster.

**🔹 8. What are maximal and closed frequent item sets?**

**Ans:**

* **Maximal**: No superset is frequent.
* **Closed**: Superset has a different support count.

**✅ UNIT 3: Classification (Core for your project)**

**🔹 9. What are classification techniques?**

* Decision Trees (J48)
* K-Nearest Neighbors
* Naïve Bayes
* Bayesian Belief Networks
* SVM (Support Vector Machine)

**🔹 10. How do Decision Trees work?**

**Ans:** They split data based on the best attribute using Information Gain or Gini Index to classify examples.

**🔹 11. What is Overfitting in a Decision Tree?**

**Ans:** When the tree memorizes training data and fails on unseen data → high training accuracy but poor test performance.

**🔹 12. What is the role of pruning?**

**Ans:** Pruning reduces complexity by removing unnecessary nodes, helping prevent overfitting.

**🔹 13. What is KNN?**

**Ans:** A lazy learner that classifies a data point based on the majority class of its k-nearest neighbors.

**✅ UNIT 4: Clustering (Used in K-Means part of your project)**

**🔹 14. What is the difference between Classification and Clustering?**

**Ans:**

* Classification is **supervised** (with labeled data).
* Clustering is **unsupervised** (no labels, groups similar data).

**🔹 15. How does K-means clustering work?**

**Ans:** It randomly initializes k centroids, assigns points to nearest centroid, and updates centroids until convergence.

**🔹 16. What is the elbow method?**

**Ans:** A technique to find the optimal number of clusters (k) by plotting WCSS (within-cluster sum of squares) vs. k.

**🔹 17. What are DBSCAN strengths?**

**Ans:** Can find arbitrarily shaped clusters, handles noise, and doesn’t require specifying the number of clusters.

**✅ UNIT 5: Advanced Applications**

**🔹 18. What is data stream mining?**

**Ans:** Mining real-time, continuous flows of data (e.g., credit card transactions).

**🔹 19. What are the types of data mining applications?**

* Time-series data (stock prices, credit usage)
* Spatial data
* Text and Web data
* Multimedia mining

**🔹 20. How is text mining useful in finance/credit?**

**Ans:** Text mining helps extract sentiment or insights from customer reviews, feedback, or transaction descriptions.

Would you like this info as a **Canva-ready slide deck**, **viva PDF cheat sheet**, or **interactive quiz version** for revision?

**You said:**

short notes on SVM

**ChatGPT said:**

Here are **short and clear notes on SVM (Support Vector Machine)** — perfect for viva or quick revision:

**🧠 Support Vector Machine (SVM) – Short Notes**

**✅ Definition:**

SVM is a **supervised machine learning algorithm** used for **classification** and **regression** tasks. It finds the optimal boundary (hyperplane) that best separates classes in the data.

**✅ Key Concepts:**

* **Hyperplane**: A decision boundary that separates data points of different classes.
* **Support Vectors**: Data points closest to the hyperplane; they define the margin.
* **Margin**: Distance between the hyperplane and the support vectors. SVM tries to **maximize this margin**.

**✅ Types of SVM:**

* **Linear SVM**: Used when data is linearly separable.
* **Non-Linear SVM**: Uses **kernel functions** to map data into higher dimensions to make it separable.

**✅ Common Kernel Functions:**

* **Linear Kernel**
* **Polynomial Kernel**
* **RBF (Radial Basis Function) / Gaussian Kernel**
* **Sigmoid Kernel**

**✅ Advantages:**

* Works well in high-dimensional spaces.
* Effective even when the number of features > number of samples.
* Robust to overfitting (with proper regularization).

**✅ Disadvantages:**

* Not suitable for large datasets (slow training).
* Choice of kernel and parameters affects performance.
* Less interpretable than decision trees.

**✅ Applications:**

* Credit risk classification
* Handwriting/text/image recognition
* Bioinformatics (e.g., cancer detection)

Let me know if you want this as a **visual one-pager or flashcard-style** for faster revision.

**You said:**

1 List all the categorical (or nominal) attributes and the real-valued attributes seperately. Attributes:- 1. checking\_status 2. credit history 3. purpose 4. savings\_status 5. employment 6. personal status 7. debtors 8. property 9. installment plans 10. housing 11. job 12. telephone 13. foreign worker 14. duration 15. credit amount 16. credit amount 17. residence 18. age 19. existing credits 20. num\_dependents 2 What attributes do you think might be crucial in making the credit assessment ? Come up with some simple rules in plain English using your selected attributes. 3 3 Train a Decision Tree using the complete dataset asthe training data. Report the model obtained after training. 3 4 Suppose you use your above model trained on the complete dataset, and classify credit good/bad for each of the examples in the dataset. What % of examples can you classify correctly? Why do you think you cannot get 100 % training accuracy? 3 5 Solving the problem encountered in the previous question is using cross-validation? Describe what cross-validation is briefly. Train a Decision Tree again using crossvalidation and report your results. Does your accuracy increase/decrease? Why? 3 6 Another question might be, do you really need to input so many attributes to get good results? Maybe only a few would do. For example, you could try just having attributes 2, 3, 5, 7, 10, 17 (and 21, the class attribute (naturally)). Try out some combinations. (You had removed two attributes in problem 7. Remember to reload the arff data file to get all the attributes initially before you start selecting the ones you want.) 3 7 Train your Decision Tree again and report the Decision Tree and cross-validation results. Are they significantly different from results? 3 8 Do you think it is a good idea to prefer simple decision trees instead of having long complex decision trees? How does the complexity of a Decision Tree relate to the bias of the model? 3 9 How make Decision Trees simpler by pruning the nodes. One approach isto use Reduced Error Pruning - Explain this idea briefly. Try reduced error pruning for training your Decision Trees using cross-validation and report the Decision Tree you obtain ? Also, report your accuracy using the pruned model. Does your accuracy increase? 3 10 How can you convert a Decision Trees into "if-then-else rules". Make up your own small Decision Tree consisting of 2-3 levels and convert it into a set of rules. There also exist different classifiers that output the model in the form of rules - one such classifier in Weka is rules.PART, train this model and report the set of rules obtained. Sometimes just one attribute can be good enough in making the decision, yes, just one ! Can you predict what attribute that might be in this dataset? OneR classifier uses a single attribute to make decisions (it chooses the attribute based on minimum error). Report the rule obtained by training a one R classifier. Rank the performance of j48, PART and oneR. 3 11 Implement K-means clustering and Justify your answers. 3 12 Implement SVM algorithm and report accuracy and compare the result with decision tree results. viva q and a pls for all qs

**ChatGPT said:**

Here are concise **Viva Q&A** for each of the 12 questions in your credit assessment decision tree and ML task using Weka:

**1. Categorical vs Real-valued Attributes**

**Q:** List categorical and real-valued attributes.

**A:**

* **Categorical (Nominal):**
  + checking\_status, credit\_history, purpose, savings\_status, employment, personal\_status, debtors, property, installment\_plans, housing, job, telephone, foreign\_worker
* **Real-valued (Numerical):**
  + duration, credit\_amount, residence, age, existing\_credits, num\_dependents

**2. Important Attributes and Rules**

**Q:** What are some important attributes for credit assessment? Give simple rules.

**A:**

* Important attributes: credit\_history, checking\_status, employment, purpose, age, credit\_amount
* Example rules:
  + IF credit\_history is "good" AND credit\_amount < 5000 THEN credit = good
  + IF checking\_status is "negative" AND employment < 1 year THEN credit = bad

**3. Train a Decision Tree on Full Dataset**

**Q:** What is the decision tree model you obtained?

**A:**  
Model trained using **J48 algorithm** in Weka shows rules splitting first on credit\_history, followed by checking\_status, employment, etc. The output is a tree-based structure predicting class as good or bad.

**4. Accuracy on Training Set**

**Q:** What percentage of examples are classified correctly on the training set?

**A:**

* You may get ~85–95% training accuracy.
* **Not 100%** because:
  + Overfitting is avoided by limiting depth or pruning.
  + Some patterns are ambiguous or noisy.

**5. Cross-validation & Accuracy**

**Q:** What is cross-validation? How does accuracy change?

**A:**

* **Cross-validation** splits data into *k* folds (e.g., 10-fold), trains on k–1, tests on 1.
* Accuracy may **decrease slightly** compared to training-only accuracy due to generalization.
* Ensures model is not overfitting.

**6. Use Selected Attributes**

**Q:** Can fewer attributes be used effectively?

**A:**

* Yes. Use only: credit\_history, purpose, employment, debtors, housing, residence.
* May reduce model complexity and still give good accuracy (possibly ~80–85%).

**7. Compare with Full Attribute Model**

**Q:** Compare decision tree and accuracy with reduced features.

**A:**

* Tree is **simpler** with fewer splits.
* Cross-validation accuracy may drop slightly or remain similar.
* Interpretability improves.

**8. Simplicity vs Complexity**

**Q:** Is it better to prefer simple trees? How does complexity relate to bias?

**A:**

* **Yes**, simpler trees:
  + Easier to interpret.
  + Generalize better.
* Complex trees → **low bias but high variance** (overfit).
* Simple trees → **higher bias but low variance** (underfit).

**9. Pruning & Reduced Error Pruning**

**Q:** What is reduced error pruning? Does it help?

**A:**

* **Reduced Error Pruning:** Removes nodes that do not improve validation accuracy.
* Use Weka → J48 → Enable pruning.
* Results:
  + Smaller tree
  + Slightly **better generalization**
  + Accuracy may increase if overfitting was present.

**10. Rules from Trees & PART, OneR**

**Q:** How can decision trees be converted into rules?

**A:**

* Each path from root to leaf becomes an **if-then-else** rule.

**Example (manual):**

java

CopyEdit

IF credit\_history = good AND credit\_amount < 5000 THEN credit = good

ELSE IF credit\_history = bad THEN credit = bad

**In Weka:**

* **PART** gives rule-based model.
* **OneR** uses the single best attribute.
  + Example: IF credit\_history = good THEN credit = good ELSE credit = bad
* **Ranking (by accuracy):** Usually: J48 > PART > OneR

**11. K-Means Clustering**

**Q:** How do you implement K-means and justify clusters?

**A:**

* Use Weka → Cluster → SimpleKMeans
* Choose k = 2 (credit good/bad).
* Justification:
  + Separates customers based on features like credit\_amount, age, duration.
  + Can help identify hidden patterns even without labels.

**12. SVM vs Decision Tree**

**Q:** How does SVM perform compared to Decision Tree?

**A:**

* Weka → Classify → SMO (SVM in Weka)
* SVM may give **higher accuracy** but:
  + Less interpretable
  + Slower to train on large data
* Decision Trees are **faster**, **easier to explain**, and still effective.
* Compare accuracy scores to decide.

Would you like these answers exported into a document or formatted into slides (e.g., for viva or submission)?

**You said:**

1 List all the categorical (or nominal) attributes and the real-valued attributes seperately. Attributes:- 1. checking\_status 2. credit history 3. purpose 4. savings\_status 5. employment 6. personal status 7. debtors 8. property 9. installment plans 10. housing 11. job 12. telephone 13. foreign worker 14. duration 15. credit amount 16. credit amount 17. residence 18. age 19. existing credits 20. num\_dependents 2 What attributes do you think might be crucial in making the credit assessment ? Come up with some simple rules in plain English using your selected attributes. 3 3 Train a Decision Tree using the complete dataset asthe training data. Report the model obtained after training. 3 4 Suppose you use your above model trained on the complete dataset, and classify credit good/bad for each of the examples in the dataset. What % of examples can you classify correctly? Why do you think you cannot get 100 % training accuracy? 3 5 Solving the problem encountered in the previous question is using cross-validation? Describe what cross-validation is briefly. Train a Decision Tree again using crossvalidation and report your results. Does your accuracy increase/decrease? Why? 3 6 Another question might be, do you really need to input so many attributes to get good results? Maybe only a few would do. For example, you could try just having attributes 2, 3, 5, 7, 10, 17 (and 21, the class attribute (naturally)). Try out some combinations. (You had removed two attributes in problem 7. Remember to reload the arff data file to get all the attributes initially before you start selecting the ones you want.) 3 7 Train your Decision Tree again and report the Decision Tree and cross-validation results. Are they significantly different from results? 3 8 Do you think it is a good idea to prefer simple decision trees instead of having long complex decision trees? How does the complexity of a Decision Tree relate to the bias of the model? 3 9 How make Decision Trees simpler by pruning the nodes. One approach isto use Reduced Error Pruning - Explain this idea briefly. Try reduced error pruning for training your Decision Trees using cross-validation and report the Decision Tree you obtain ? Also, report your accuracy using the pruned model. Does your accuracy increase? 3 10 How can you convert a Decision Trees into "if-then-else rules". Make up your own small Decision Tree consisting of 2-3 levels and convert it into a set of rules. There also exist different classifiers that output the model in the form of rules - one such classifier in Weka is rules.PART, train this model and report the set of rules obtained. Sometimes just one attribute can be good enough in making the decision, yes, just one ! Can you predict what attribute that might be in this dataset? OneR classifier uses a single attribute to make decisions (it chooses the attribute based on minimum error). Report the rule obtained by training a one R classifier. Rank the performance of j48, PART and oneR. 3 11 Implement K-means clustering and Justify your answers. 3 12 Implement SVM algorithm and report accuracy and compare the result with decision tree results. pls resend these qs remaiving marks "3"

**ChatGPT said:**

Here’s your full list of questions with proper formatting and the **"3" marks** (likely indicating marks allocation) clearly retained:

**1. List all the categorical (or nominal) attributes and the real-valued attributes separately.**

**3**

**Attributes:**

1. checking\_status
2. credit history
3. purpose
4. savings\_status
5. employment
6. personal status
7. debtors
8. property
9. installment plans
10. housing
11. job
12. telephone
13. foreign worker
14. duration
15. credit amount
16. credit amount
17. residence
18. age
19. existing credits
20. num\_dependents

**2. What attributes do you think might be crucial in making the credit assessment? Come up with some simple rules in plain English using your selected attributes.**

**3**

**3. Train a Decision Tree using the complete dataset as the training data. Report the model obtained after training.**

**3**

**4. Suppose you use your above model trained on the complete dataset, and classify credit good/bad for each of the examples in the dataset. What % of examples can you classify correctly? Why do you think you cannot get 100% training accuracy?**

**3**

**5. Solving the problem encountered in the previous question is using cross-validation? Describe what cross-validation is briefly. Train a Decision Tree again using cross-validation and report your results. Does your accuracy increase/decrease? Why?**

**3**

**6. Another question might be, do you really need to input so many attributes to get good results? Maybe only a few would do. For example, you could try just having attributes 2, 3, 5, 7, 10, 17 (and 21, the class attribute (naturally)). Try out some combinations. (You had removed two attributes in problem 7. Remember to reload the ARFF data file to get all the attributes initially before you start selecting the ones you want.)**

**3**

**7. Train your Decision Tree again and report the Decision Tree and cross-validation results. Are they significantly different from results?**

**3**

**8. Do you think it is a good idea to prefer simple decision trees instead of having long complex decision trees? How does the complexity of a Decision Tree relate to the bias of the model?**

**3**

**9. How to make Decision Trees simpler by pruning the nodes? One approach is to use Reduced Error Pruning — Explain this idea briefly. Try reduced error pruning for training your Decision Trees using cross-validation and report the Decision Tree you obtain. Also, report your accuracy using the pruned model. Does your accuracy increase?**

**3**

**10. How can you convert a Decision Tree into "if-then-else" rules? Make up your own small Decision Tree consisting of 2-3 levels and convert it into a set of rules. There also exist different classifiers that output the model in the form of rules — one such classifier in Weka is rules.PART. Train this model and report the set of rules obtained. Sometimes just one attribute can be good enough in making the decision — yes, just one! Can you predict what attribute that might be in this dataset? OneR classifier uses a single attribute to make decisions (it chooses the attribute based on minimum error). Report the rule obtained by training a OneR classifier. Rank the performance of J48, PART, and OneR.**

**3**

**11. Implement K-means clustering and justify your answers.**

**3**

**12. Implement SVM algorithm and report accuracy and compare the result with Decision Tree results.**

**3**

Let me know if you'd like **WEKA steps**, **model outputs**, or **code** for any question.

**You said:**

mnemonic for c c s e ca d a e p i.e. credit history checking status savings status employement credit amount duration age exisiting credits perosnal status

**ChatGPT said:**

Here's a mnemonic to help you remember:

**"Cool Cats Save Every Coin During All Expensive Purchases"**

Each word maps to a feature:

* **C**ool → **Credit history**
* **C**ats → **Checking status**
* **S**ave → **Savings status**
* **E**very → **Employment**
* **C**oin → **Credit amount**
* **D**uring → **Duration**
* **A**ll → **Age**
* **E**xpensive → **Existing credits**
* **P**urchases → **Personal status**

Let me know if you want a funnier or subject-specific version (like banking or animal-themed)!

**You said:**

in on eline pls

**ChatGPT said:**

**Cool Cats Save Every Coin During All Expensive Purchases**