7) Train your decision Tree again & report the Decision Tree and cross validation results. Are they significantly different from results?

* Decision Tree result:-

	TP Rate	FP Rate	Precision	Recall	F-M	Class
	0.956	0.380	0.854	0.956	0.902	good
	0.620	0.044	0.857	0.620	0.720	bad
weighted avg.	0.855	0.279	0.855	0.855	0.847	

Confusion Matrix :-

a	b
669	31
114	186

* Cross validation result:- 10 folds.

	TP Rate	FP Rate	Precision	Recall	F-M	Class
	0.840	0.610	0.763	0.840	0.799	good
	0.390	0.160	0.511	0.390	0.442	bad
weighted AVG	0.705	0.475	0.687	0.705	0.692	

Confusion Matrix

a	b
588	112
183	117

• Yes, they are different.

• The precision in cross validation decreases.

• Cross validation gives more reliable estimate of real-world performance, while the single model tree precision is overly optimistic.

8) Do you think it is o
simple decision trees inste
decision trees? How do
Tree relate to the bi

• Yes, it is generally
simple decision tree
Because, simpler tre
explain and maintain
when a tree bec
capturing noise.

* Complexity & Model

• Complex Decision
As a tree beco
with more bra
data very do
due to oversimpl
Model becomes
the data and
new Data.

• Simpler Decision
Simpler trees
may not cap
but they have
robust to
This typically

) 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?

- Yes, it is generally a good idea to prefer simple decision tree over long, complex ones. Because, simpler trees are easier to interpret, explain and maintain. whereas overfitting occurs when a tree becomes too complex & starts capturing noise.

* complexity & Model Bias

- Complex Decision Tree:-

As a tree becomes more complex (deeper with more branches.) it can fit the training data very closely, reducing bias (error due to oversimplification).

Model becomes sensitive to small changes in the data and may not perform well on new Data.

- Simpler Decision Tree:-

Simpler trees having higher bias because they may not capture all the patterns in the data, but they have lower variance and are more robust to new, unseen data.

This typically leads to generalisation.

	TP Rate	FP Rate	Precision	Recall	F-M	class
	0.940	0.623	0.779	0.940	0.852	good
	0.377	0.060	0.729	0.377	0.497	bad
weighted avg	0.771	0.454	0.764	0.771	0.745	

confusion matrix

a	b
658	42
187	113

Selecting 3 attributes :- checking status, duration, class

	TP Rate	FP Rate	Precision	Recall	F M	class
	0.767	0.370	0.829	0.767	0.797	good
	0.630	0.233	0.537	0.630	0.580	bad
weighted avg	0.726	0.329	0.74	0.725	0.732	

confusion matrix

a	b
537	163
111	189

Selection of 1st attribute :- checking status, class to

	TP Rate	FP Rate	Precision	Recall	FM	class
	0.957	0.377	0.856	0.957	0.904	good
	0.623	0.043	0.862	0.623	0.723	bad
weighted avg	0.857	0.277	0.858	0.857	0.850	

confusion Matrix :-

a	b
670	30
113	187

26/06/25

b) 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.)

* No, you really do not always need to use all available attributes to get good results. Often, a smaller, well chosen subset of attributes can provide similar or even better performance especially if the excluded attributes are irrelevant or noisy.

* For 21 attribute

	TP Rate	Precision	FP Rate	Recall	F-M	class
	0.841	0.764	0.607	0.841	0.801	good
	0.393	0.515	0.159	0.393	0.446	bad
weighted avg	0.707	0.687	0.472	0.707	0.694	

* For 21 attribute :- checking status, class.

	TP Rate	FP Rate	Precision	Recall	F-M	Class
	1.000	1.000	0.700	1.000	0.824	good
	0.000	0.000	?	0.000	?	bad
weighted avg	0.700	0.700	?	0.7000	?	

Confusion matrix :-

a	b	
700	0	good = a
300	0	bad = b

Selecting 10 attributes . checking status, duration, credit history, purpose, credit-amount saving-status, employment property-magnitude, age, class

TP Rate	FP Rate	Precision	Recall	F-M	class
0.897	0.433	0.828	0.897	0.861	good
0.567	0.103	0.702	0.567	0.627	bad
weighted 0.798	0.334	0.791	0.798	0.791	

Confusion matrix

a	b
628	72
130	170

Selecting 7 attributes :- checking-status, duration, credit history, purpose credit-amount, employment, class

TP Rate	FP Rate	Precision	Recall	F-M	class
0.950	0.523	0.809	0.950	0.874	good
0.477	0.050	0.805	0.477	0.598	bad
weighted Avg 0.868	0.381	0.807	0.808	0.791	

Confusion Matrix

a	b
665	35
157	143

Selecting 5 attributes :- checking-status, duration, credit history, purpose, ~~tred~~ class .

TP Rate	FP Rate	Pr
0.940	0.623	0.
0.377	0.060	0
weighted avg 0.771	0.454	0.

confusion matrix

a	b
658	
187	

Selecting 3 Attributes :

TP Rate	FP Rate
0.767	0.370
0.630	0.23
weighted avg 0.726	0.32

confusion matrix

a	
537	16
111	1

Selecting 17 attributes

TP Rate	FP Rate
0.957	0.3
0.623	0.0
weighted Avg 0.857	0.27

confusion Matrix :-

0.008875	other_payment_plans
0.006811	personal_status
0.005823	foreign_worker
0.004797	other_parties
0.001337	job
0.000964	own_telephone
0	num_dependents
0	installment_commitment
0	residence_since
	existing_credits

Step2: Visualize the plot matrix

the class is classified → good [blue]
→ bad [red]

Step 3: Selecting only the first 10 attributes considering information gain in descending order.

Before selecting 10 attributes

	TP Rate	FP Rate	Precision	Recall	FM	class
	0.841	0.764	0.607	0.841	0.801	good
	0.393	0.515	0.159	0.393	0.446	bad
weighted avg	0.707	0.684	0.472	0.707	0.694	

Confusion matrix

a	b
589	111
182	118

After selecting
TP Rate
0.850
0.430
weighted avg 0.724

Confusion m

After sele
TP Rate
0.876
0.393
weighted avg 0.731

Confusion

=> Precision
If good a
If high

3) Train a
dataset
model

1) Downlo

2) Load

19/8/25

1) list all the categorical (or nominal) attributes & real valued attributes separately.

i) checking status: nominal
ii) duration: numeric ✓
iii) credit history: nominal
iv) purpose: nominal
v) credit_amount: numeric ✓
vi) saving_status: nominal
vii) employment nominal
viii) installment_commitment: numeric ✓
ix) personal_status: nominal
x) other_parties: nominal
xi) residence_since: numeric ✓
xii) property_magnitude nominal
xiii) age: numeric ✓
xiv) other_payment_plans: nominal
xv) housing: nominal
xvi) existing_credits: numeric ✓
xvii) job: nominal
xviii) num_dependents: numeric ✓
xix) own_telephone: nominal
xx) foreign_work: nominal
xxi) class: nominal

Nominal:
checking_status
credit_history
purpose
saving_status
employment
personal_status
other_parties
property_magnitude
other_payment_plans
housing
job
own_telephone
foreign_work.
class

Numeric: duration
- credit_amount
installment_commitment
residence_since
age
existing_status
num_independents

2) What attributes do you think might be crucial in making the credit assessment? Come up with simple English using selected attributes

2) Using Information Gain - - In Select attributes find infogainattribute Eval

I G	attribute
0.094739	checking_status
0.043618	credit_history
0.0329	duration
0.028115	saving_status
0.024894	purpose
0.018709	credit_amount
0.01698	property_magnitude
0.013102	employment
0.012753	housing
0.011278	age

5) Solving the problem encountered in the previous questions i.e. 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?

* Cross Validation :- It is a technique used to evaluate the performance of a machine learning model on unseen data. It helps to ensure that the model generalizes well.

Repeating the same 6 steps mentioned previously, + Cross validation = 10 folds

TP Rate	FP Rate	Precision	Recall	F-Measure	Class
0.852	0.471	0.841	0.852	0.847	good
0.529	0.148	0.551	0.529	0.540	bad
Weig. Avg 0.770	0.388	0.767	0.770	0.768	

=> When taken 10 folds for cross validation accuracy decreases because :-

i) overfitting :- on full training set accuracy is 100%. on training dataset but is not realistic for unseen data

19/6/25

ii) cross-validation tests generalization :- Each fold is tested on data it is not seen before in each fold

6) Another question ... you really need ... get good ... would do. ... having attributes ... attributes (naturally)) ... had removed ... to reload the ... attributes initi... the ones yo...

* No, you ... all available ... Often, a m... can provide ... especially ... or noisy

* For 21 ...

TP Rate	
0.841	
0.393	
weighted avg 0.707	

* For 21 att...

TP Rate	
1.000	
0.000	
weighted avg 0.900	

Confusion ma...

4) Suppose you use your above model trained on the complete dataset. What % of examples can you classify correctly? Why do you think you cannot get 100% accuracy?

Splitting the data on training data & test data.

6) i) Percentage split : 60/40

	Precision	Recall	F Measure	Class
	0.792	0.829	0.810	good
	0.462	0.402	0.430	bad
weighted avg	0.703	0.715	0.708	

ii) Percentage split : 70/30

	Precision	Recall	F-Measure	Class
	0.793	0.870	0.829	good
	0.5[illegible]	0.367	0.423	bad
weighted avg	0.7[illegible]4	0.737	0.722	

iii) Percentage split :- 80/20

	Precision	Recall	F-Measure	Class
	0.846	0.852	0.847	good
	0.551	0.529	0.540	bad
Weighted Avg	0.767	0.770	0.768	

⇒ 100% of examples can be classified

⇒ 100% training accuracy cannot be obtained because :-

overfitting

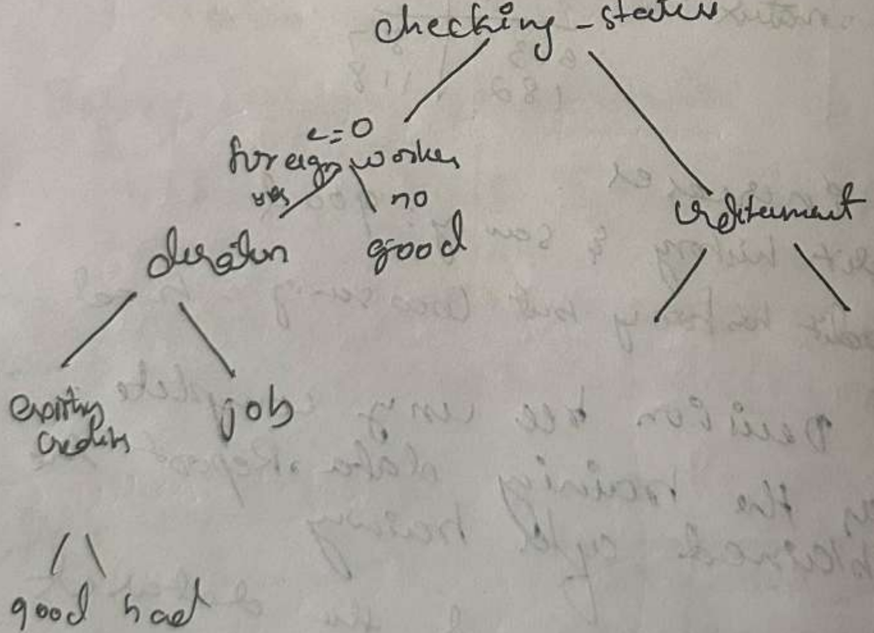
⇒ doesn't work with good accuracy on new/unseen dataset.

3) Preprocess: -

filters
↳ unsupervised
↳ attribute
↳ normalise
↳ apply to dataset

4) Using Classifier :- Using J-48 classifier for decision tree, a decision tree is formed

5) The following decision trees are made to classify good or bad. to



Observation/Report :-

- Model performs perfect on training data.
- lead to overfitting

After selecting 10 attributes

	TP Rate	FP Rate	Precision	Recall	F-1	Class
	0.850	0.570	0.773	0.850	0.812	good
	0.430	0.150	0.551	0.430	0.483	bad
weighted Avg	0.724	0.444	0.709	0.724	0.713	

Confusion matrix :-

a	b
595	105
171	129

After selecting 5 attributes

	TP Rate	FP Rate	Precision	Recall	F-M	class
	0.876	0.607	0.771	0.876	0.820	good
	0.393	0.124	0.516	0.39	0.467	bad
weighted Avg	0.731	0.462	0.712	0.71	0.714	

Confusion matrix :-

a	b
613	87
182	118

=> Precision increases.
If good credit history & saving : good
If ~~bad~~ high credit history but low saving : bad

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

1) Download :- Download the dataset

2) Load the dataset :- Load the dataset in the weka application