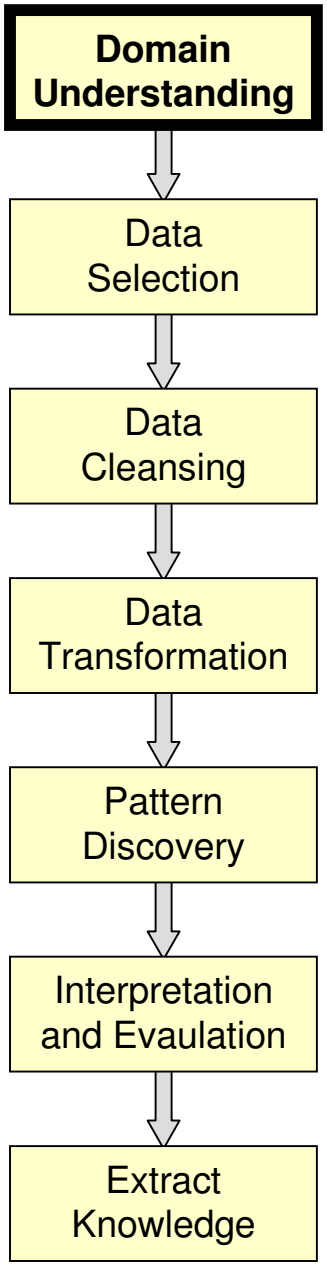
Root Cause Analysis: Shipping Plan Non-Compliance

by

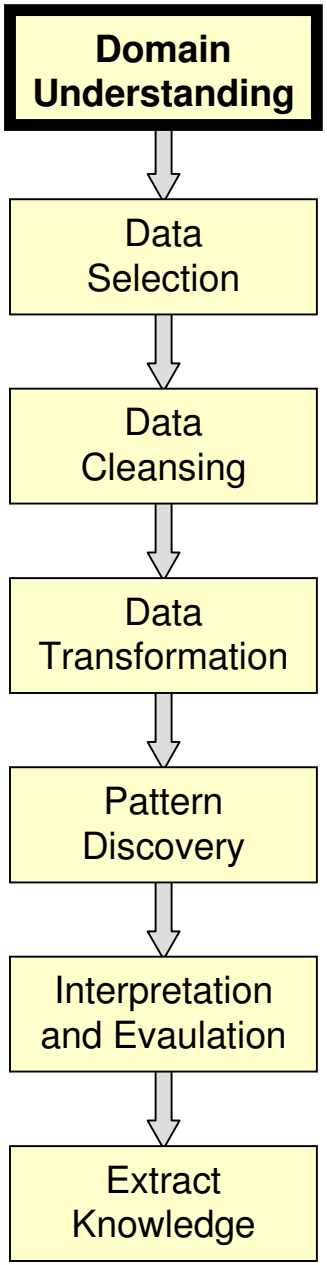
Bill Qualls

IS567 – Knowledge Discovery Technologies
DePaul University
Dr. Daniela Stan Raicu, Instructor
Winter 2011



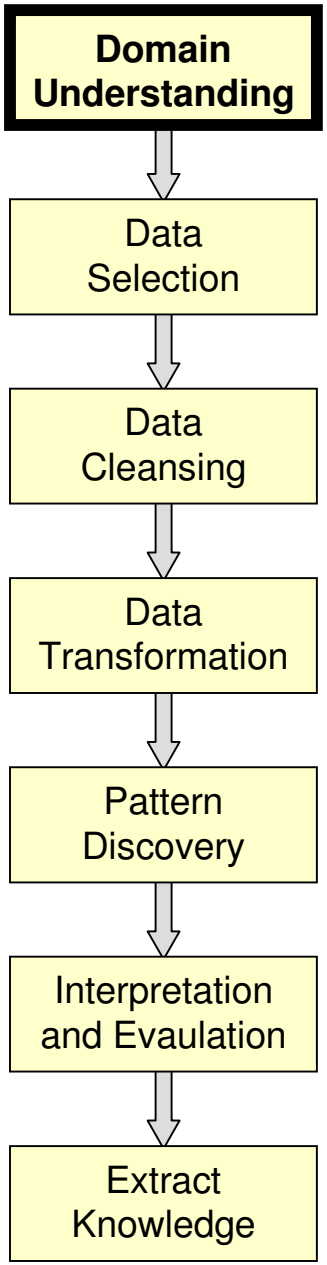
SCM meets PA

- "Customarily, supply chains have focused on day-to-day operations: The demand is forecasted, materials are sourced to meet that demand, production plans are created based on available manufacturing assets, and then produced material is shipped per requirement. The focus is on execution, not on improving decision-making, and execution is what the traditional SCM and ERP systems are meant to do." (emphasis added)



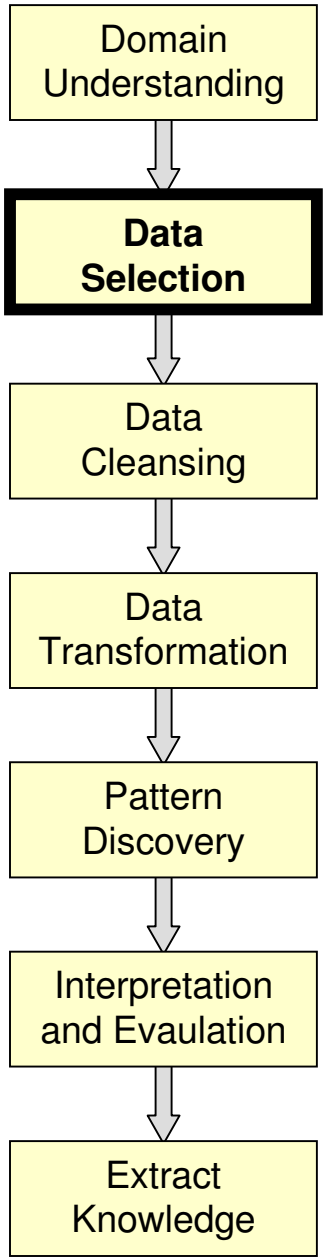
SCM meets PA

- "With predictive analytics... manufacturers can leverage the data within their legacy systems to derive real-time insights and use that information to optimize future decisions: What will be the impact of increasing prices on demand? Which promotion will have a higher impact on sales? We have limited production capacity - should we produce more of Product A or more of product B to meet our revenue and margin goals?"[1]

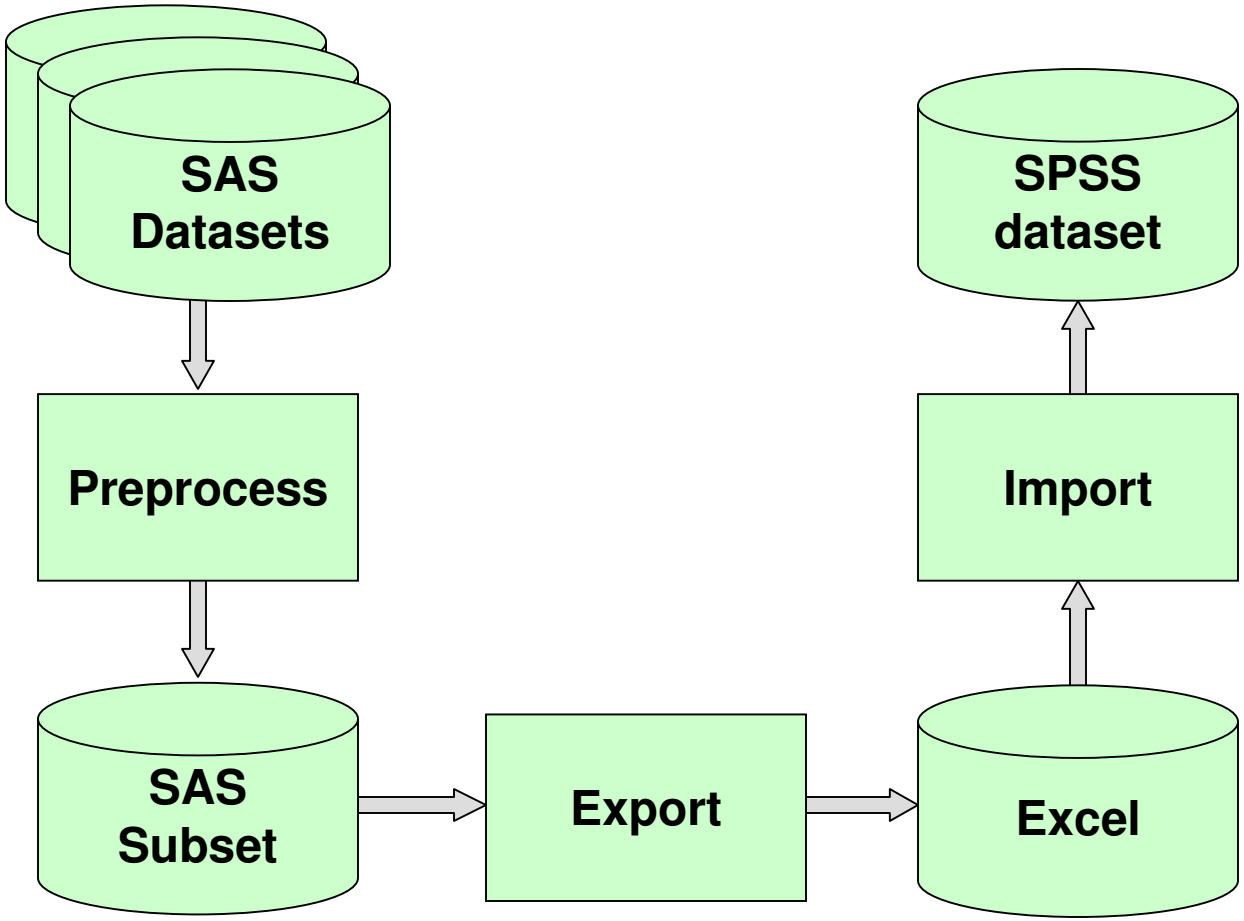


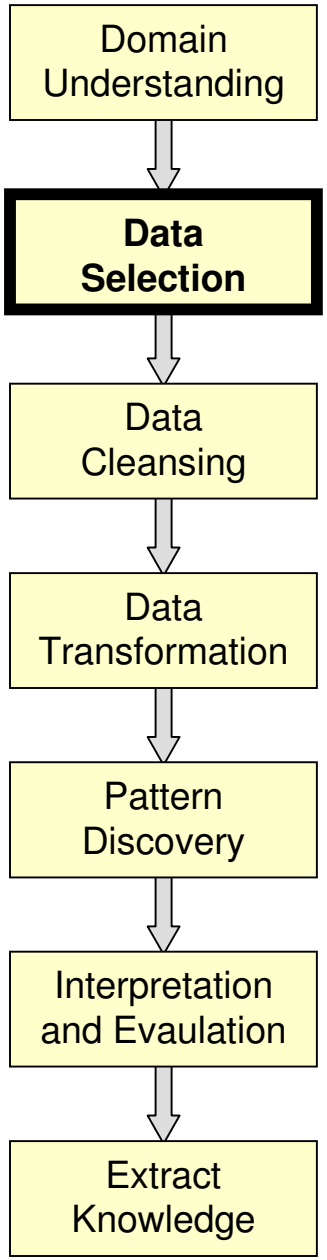
Domain Understanding

- Problem: My client's plants are not shipping product in compliance with their shipping plans.
- Client's measure of choice is **Mean Absolute Percent Error** (MAPE) defined as: abs(plan - actual) / plan, a commonly used measure of forecast accuracy.
- My goal is to **identify relationships in the data** which might be **root cause indicators** of plan compliance and non-compliance.



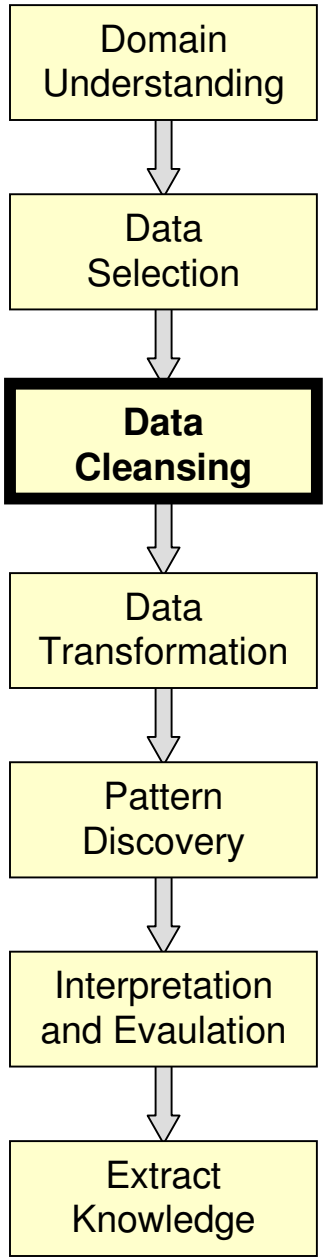
Data Selection





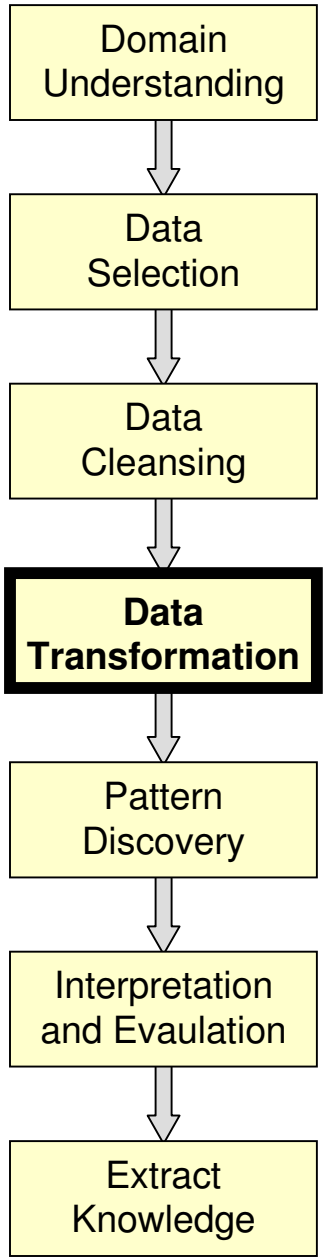
Data Selection

- Source plant
- Source plant type (Plant, Copacker, DC)
- Source plant business unit
- Source plant region
- Destination plant
- Destination plant type
- Material group
- Shipping mode (Truck, Rail)
- Product volume ranking (ABCD)
- Week (1-6)
- Item
- Pallets planned
- Pallets shipped
- MAPE ← dependent variable



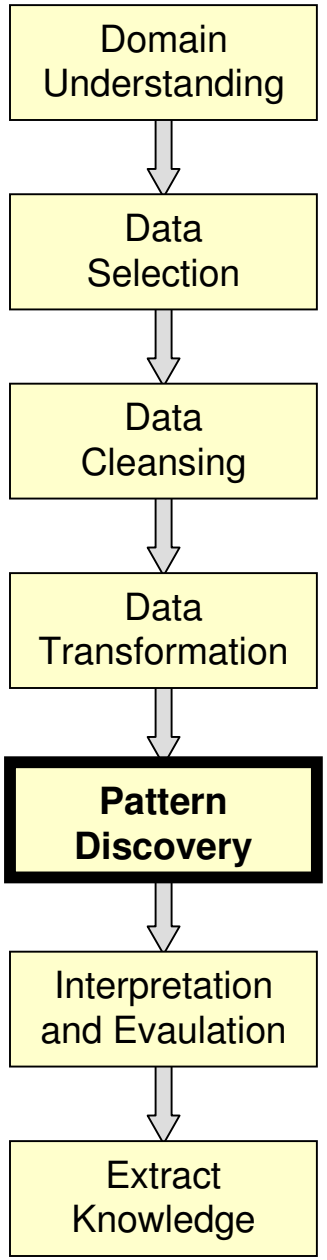
Data Cleansing

- Data cleansing was performed in SAS programs which created the data used in this process.
- Including
  - filter raw materials
  - filter shipments with negative amounts
  - filter preliminary plans



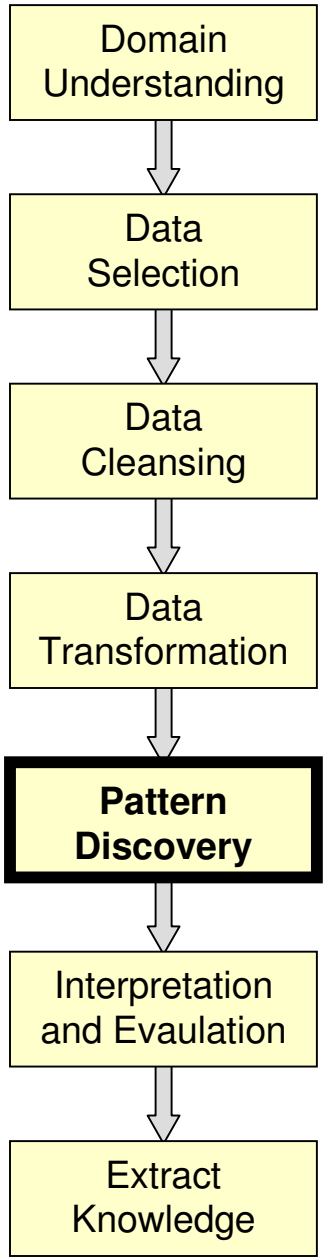
Data Transformation

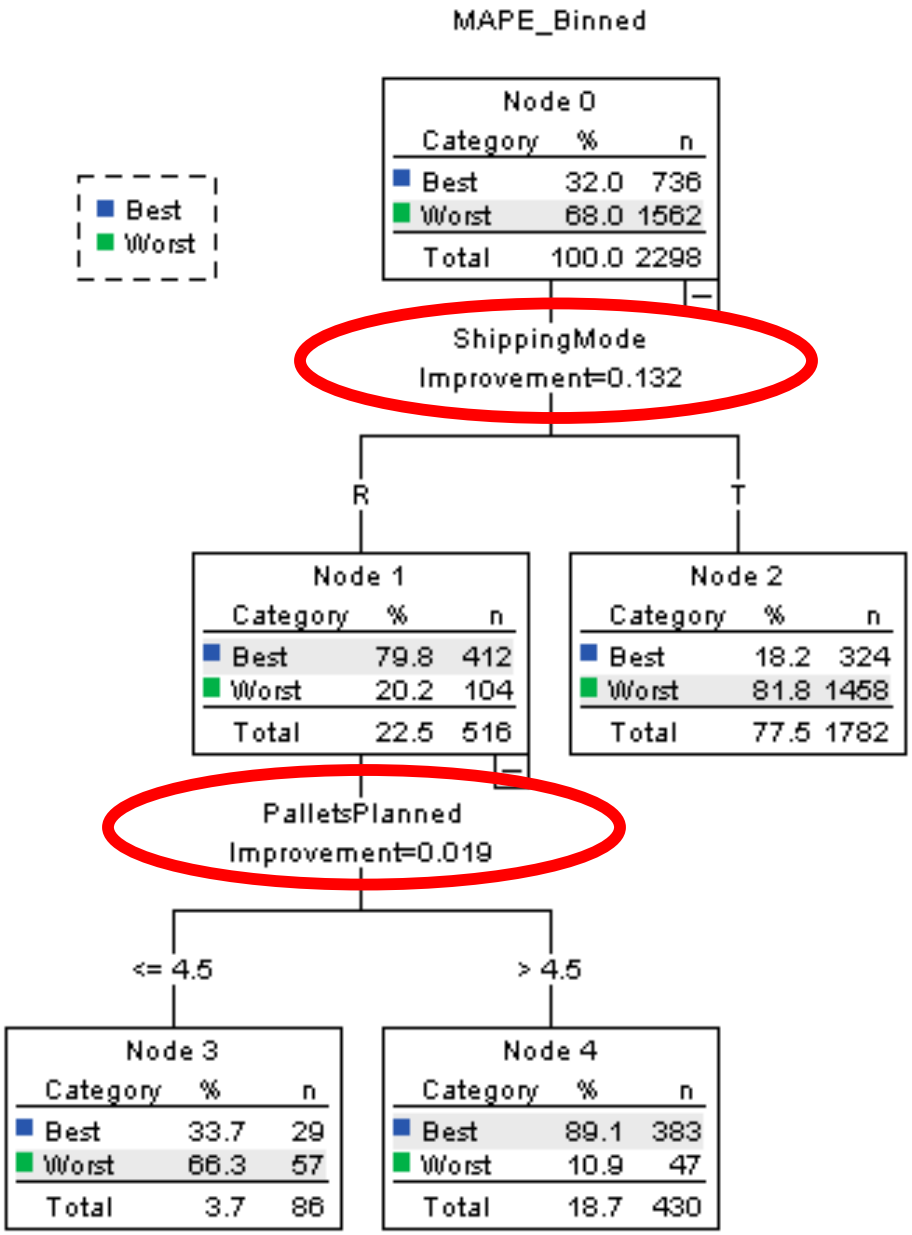
- About MAPE:
  - MAPE = undefined (missing) when plan = 0
  - MAPE = 0 when plan = actual
  - MAPE > 0 when actual < plan
  - MAPE = 1 when actual = 0 or twice the plan
  - MAPE > 1 when actual less than half of plan
- To get clean classifications I used:
  - If (0 <= MAPE<= .2) → "Best"
  - If (MAPE >= .8) → "Worst"



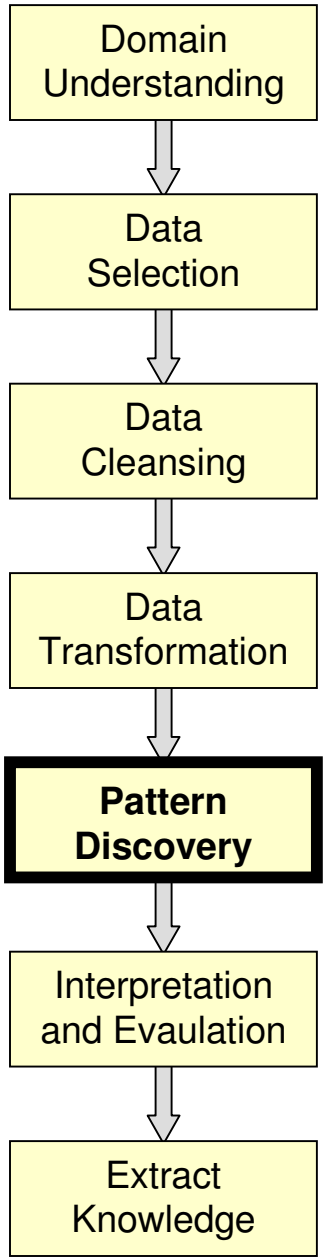
Pattern Discovery

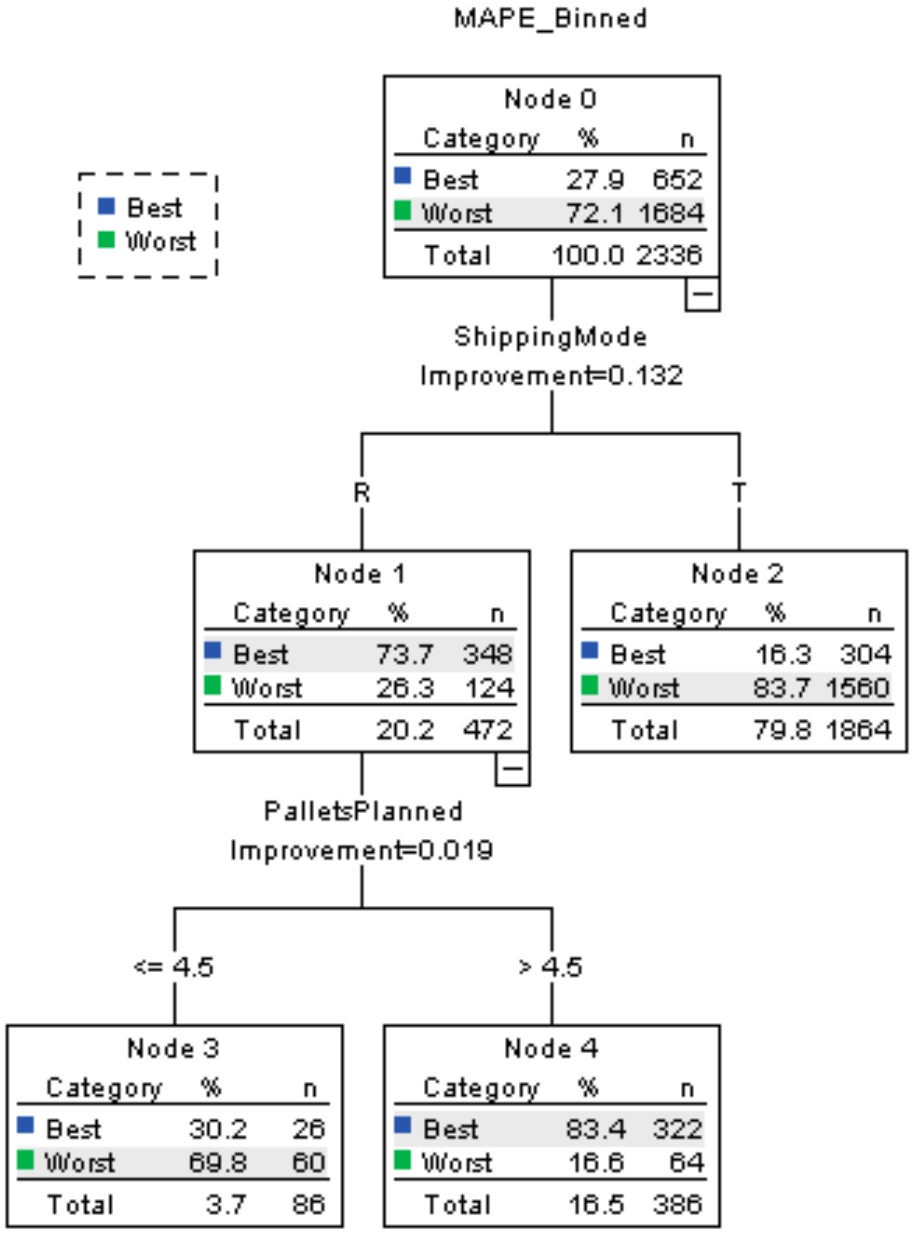
- I used **decision trees** in SPSS.
- I had chosen six weeks of data: two weeks from each of September, October, and November 2010. WEEK = 1 thru 6.
- I used weeks 1, 3, and 5 for **training**, and weeks 2, 4, and 6 for **testing**: RECODE Week (1=1) (2=2) (3=1) (4=2) (5=1) (6=2) INTO Train_Test.
- To minimize overfitting and make the resulting trees easier to understand, I set **Minimum Change in Improvement** to .01


Domain Understanding
Data Selection
Data Cleansing
Data Transformation
Pattern Discovery
Interpretation and Evaulation
Extract Knowledge

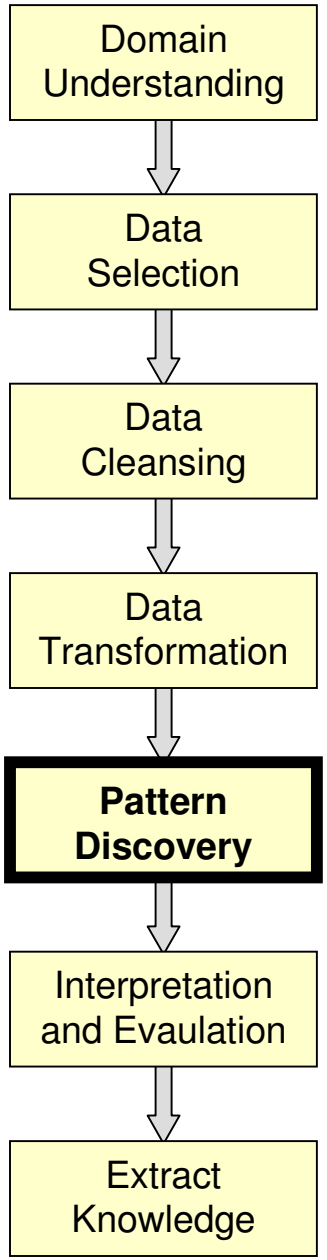

MAPE_Binned
Best
Worst
Node 0
Category % n
Best 32.0 736
Worst 68.0 1562
Total 100.0 2298
ShippingMode
Improvement=0.132
R
T
Node 1
Category % n
Best 79.8 412
Worst 20.2 104
Total 22.5 516
Node 2
Category % n
Best 18.2 324
Worst 81.8 1458
Total 77.5 1782
PalletsPlanned
Improvement=0.019
<= 4.5
> 4.5
Node 3
Category % n
Best 33.7 29
Worst 66.3 57
Total 3.7 86
Node 4
Category % n
Best 89.1 383
Worst 10.9 47
Total 18.7 430

Training


Domain Understanding
Data Selection
Data Cleansing
Data Transformation
Pattern Discovery
Interpretation and Evaulation
Extract Knowledge


MAPE_Binned
Best
Worst
Node 0
Category % n
Best 27.9 652
Worst 72.1 1684
Total 100.0 2336
ShippingMode
Improvement=0.132
R
T
Node 1
Category % n
Best 73.7 348
Worst 26.3 124
Total 20.2 472
Node 2
Category % n
Best 16.3 304
Worst 83.7 1560
Total 79.8 1864
PalletsPlanned
Improvement=0.019
<= 4.5
> 4.5
Node 3
Category % n
Best 30.2 26
Worst 69.8 60
Total 3.7 86
Node 4
Category % n
Best 83.4 322
Worst 16.6 64
Total 16.5 386

Testing

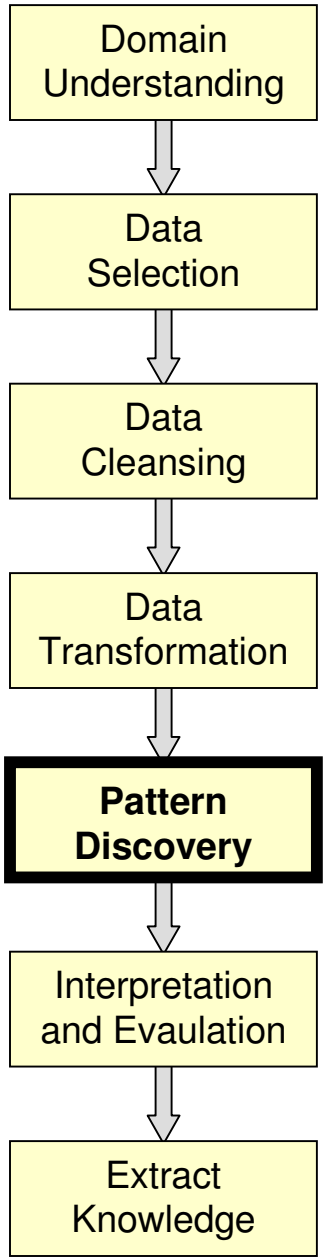


Classification

Sample	Observed	Predicted Best	Predicted Worst	Predicted Percent Correct
Training	Best	383	353	52.0%
	Worst	47	1515	97.0%
	Overall Percentage	18.7%	81.3%	82.6%
Test	Best	322	330	49.4%
	Worst	64	1620	96.2%
	Overall Percentage	16.5%	83.5%	83.1%

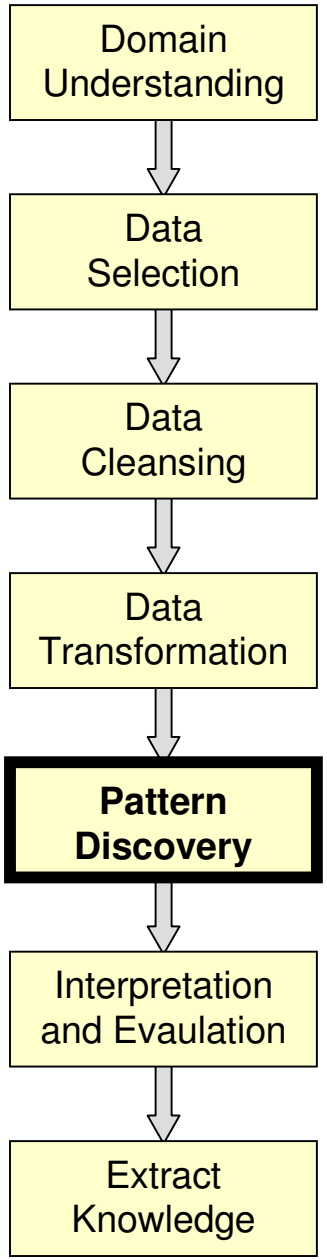
Growing Method: CRT
Dependent Variable: MAPE_Binned

Result was the same for p=100 c=50, p=50 c=25, p=10 c=5, and p=2 c=1.

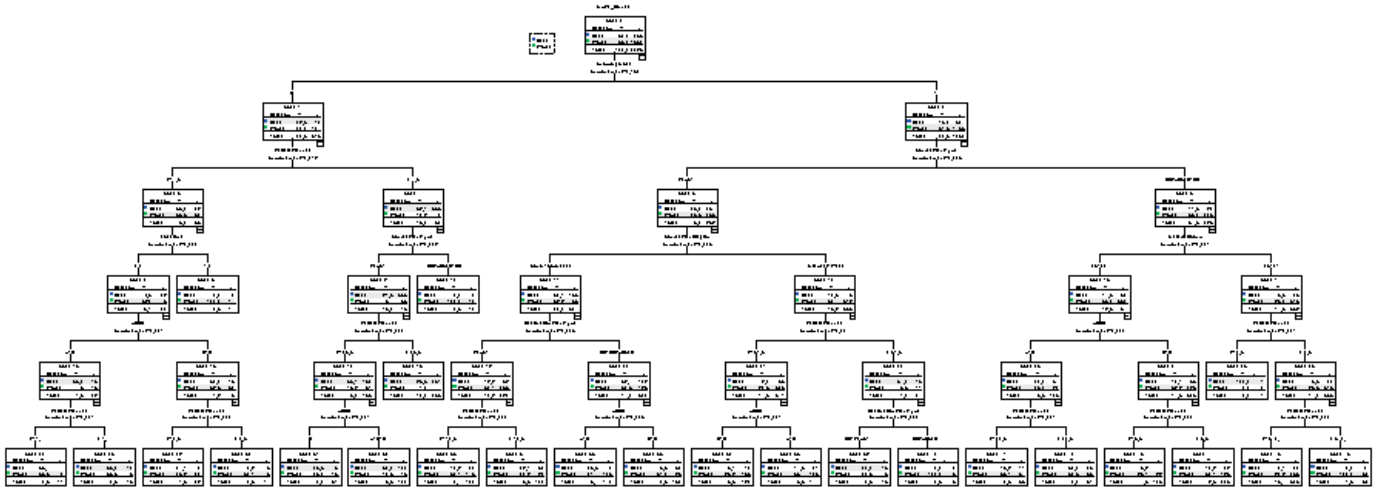


Confidence Interval

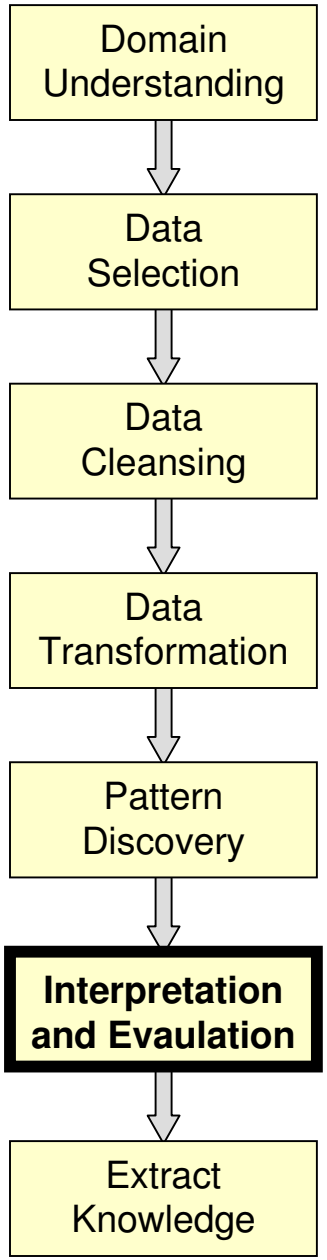
- We can construct a 95% confidence interval for the accuracy as follows:
- Classifier Error Rate (E) = number of test set errors divided by number of test set instances = (330 + 64) / 2336 = .169
- Sample variance (V) = E * (1-E) = .169 * .831 = .1404
- Standard error (SE) = sqrt(V/n) = sqrt(.1404/2336) = .00775
- Confidence interval = E ± 2*SE = .169 ± 2*.00775 = .169 ± .016 = **(.153, .185)**



Overfitting!

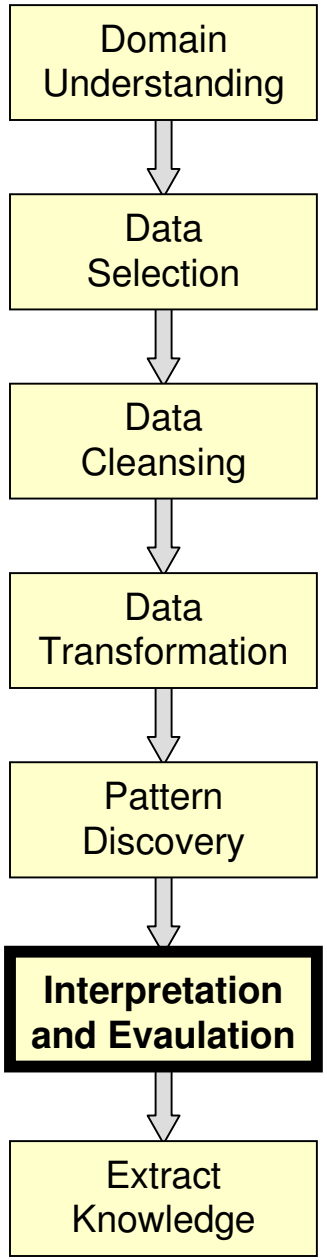


This is what happened when I used p=2 c=1 and used the default Minimum Change for Improvement (.0001)!



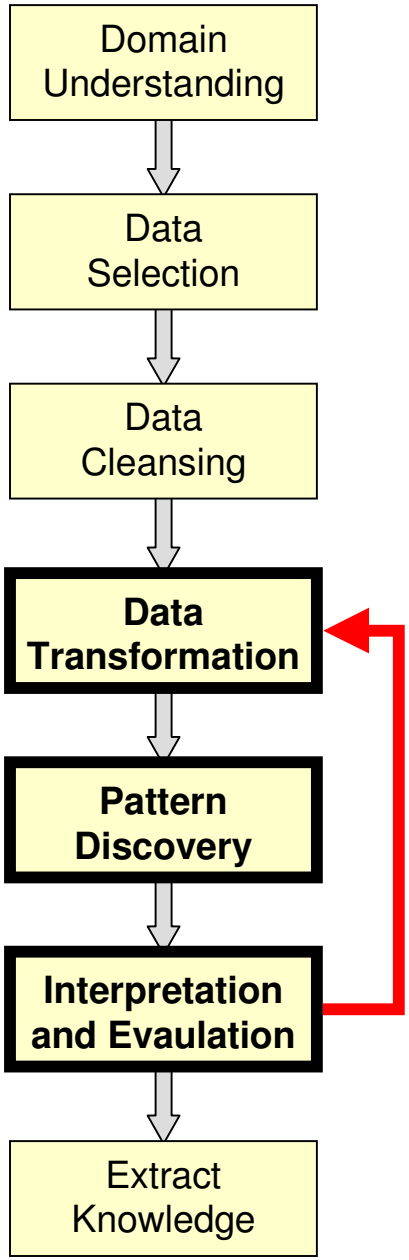
Interpretation and Evaluation

- I was not satisfied with the trivial results and attributed them to the data being too complex; that is, representing too many populations.
- I used Excel to run a **Pareto analysis** on the data. It turns out that those items with planned shipment of 50 or more constitute 80% of all shipments.



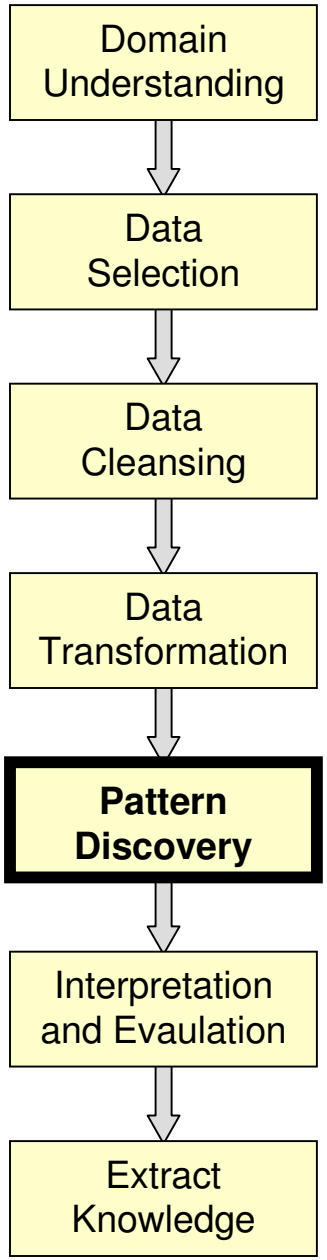
Interpretation and Evaluation

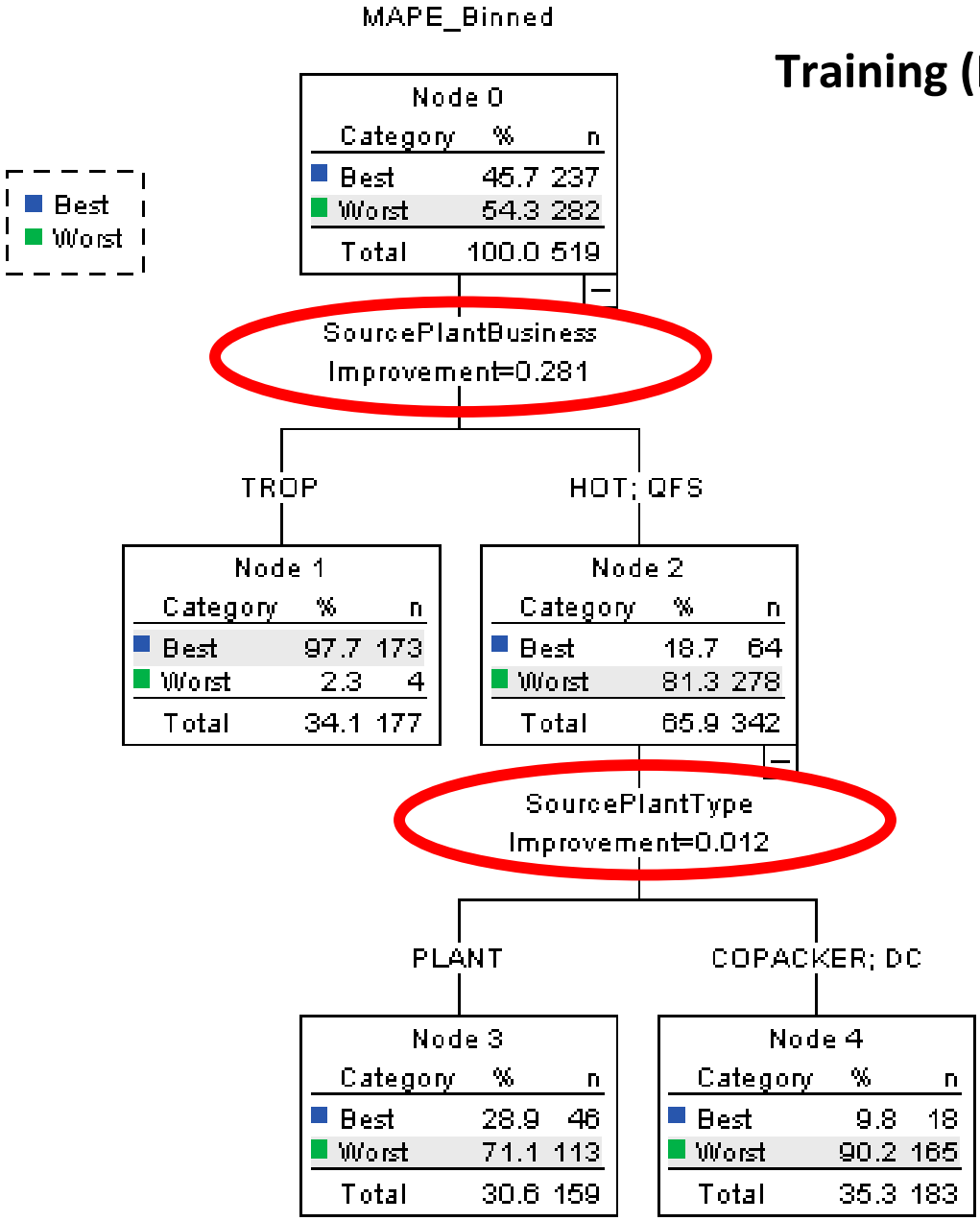
- An observation: I have 9,233 rows of data. There are 1,622 rows with planned shipment of 50 or more: that is 1,622 / 9,233 = 17.6%, which is pretty close to the proverbial 80/20 rule often mentioned with Pareto analysis!



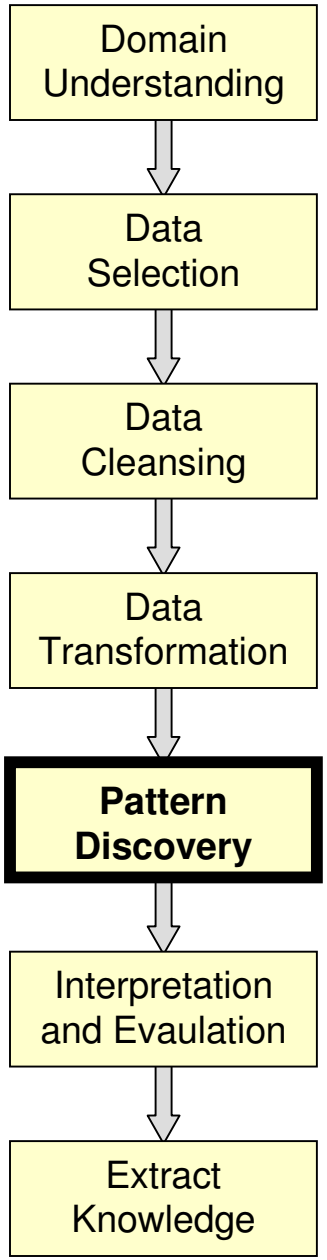
Interpretation and Evaluation

- I added a new variable indicating which rows had planned shipment of 50 or more and which did not.
- I then reran the decision tree for each condition.
- And so we see that data mining is, in fact, an iterative process!

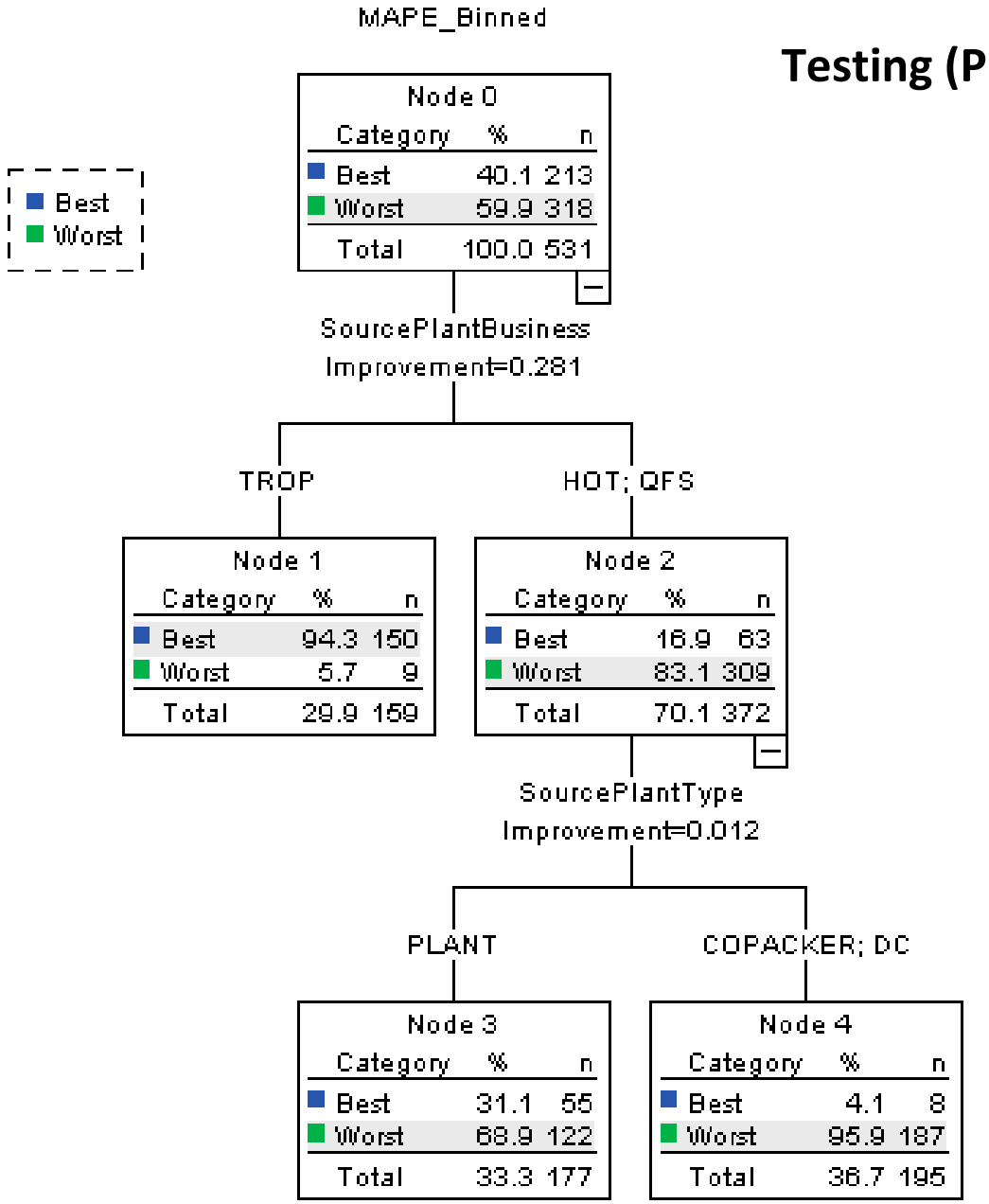


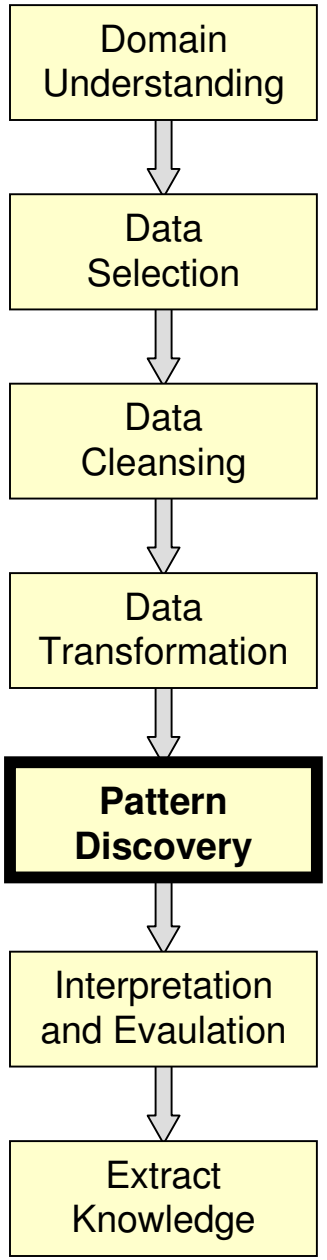


Training (Plan >= 50)



Testing (Plan >= 50)

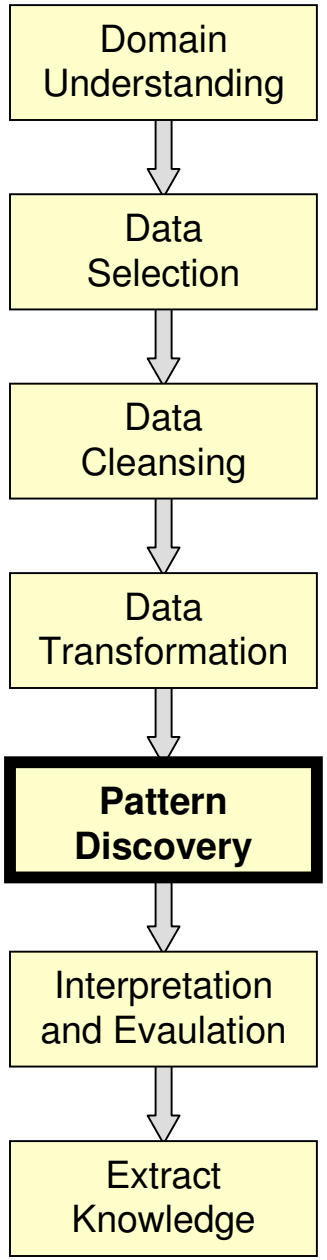


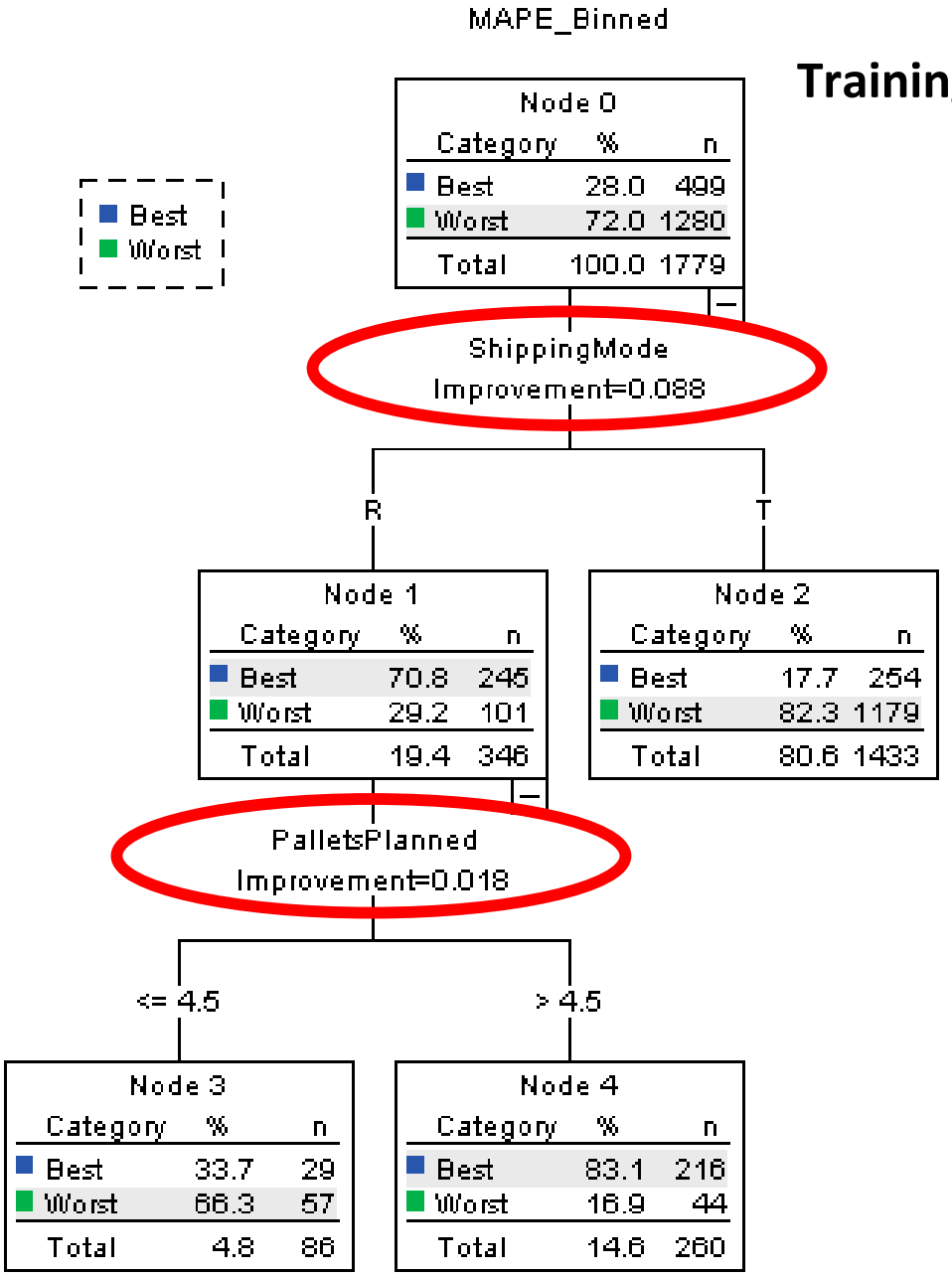


Classification (Plan > =50)

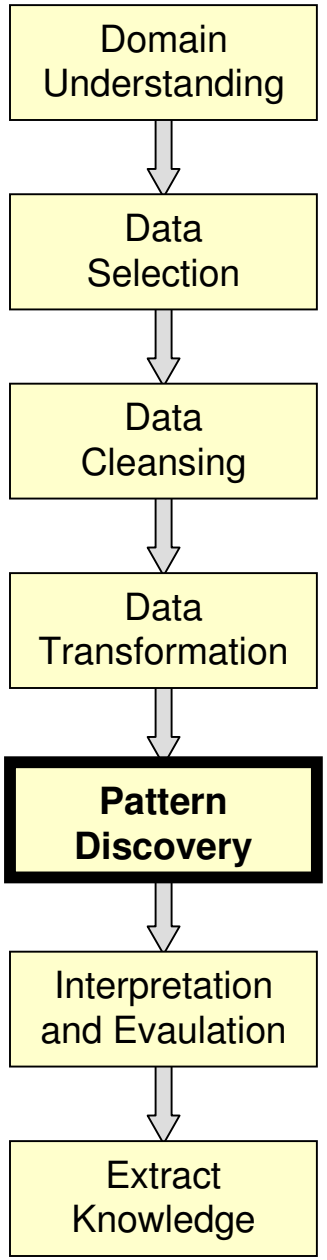
Sample	Observed	Predicted		
		Best	Worst	Percent Correct
Training	Best	173	64	73.0%
	Worst	4	278	98.6%
	Overall Percentage	34.1%	65.9%	86.9%
Test	Best	150	63	70.4%
	Worst	9	309	97.2%
	Overall Percentage	29.9%	70.1%	86.4%

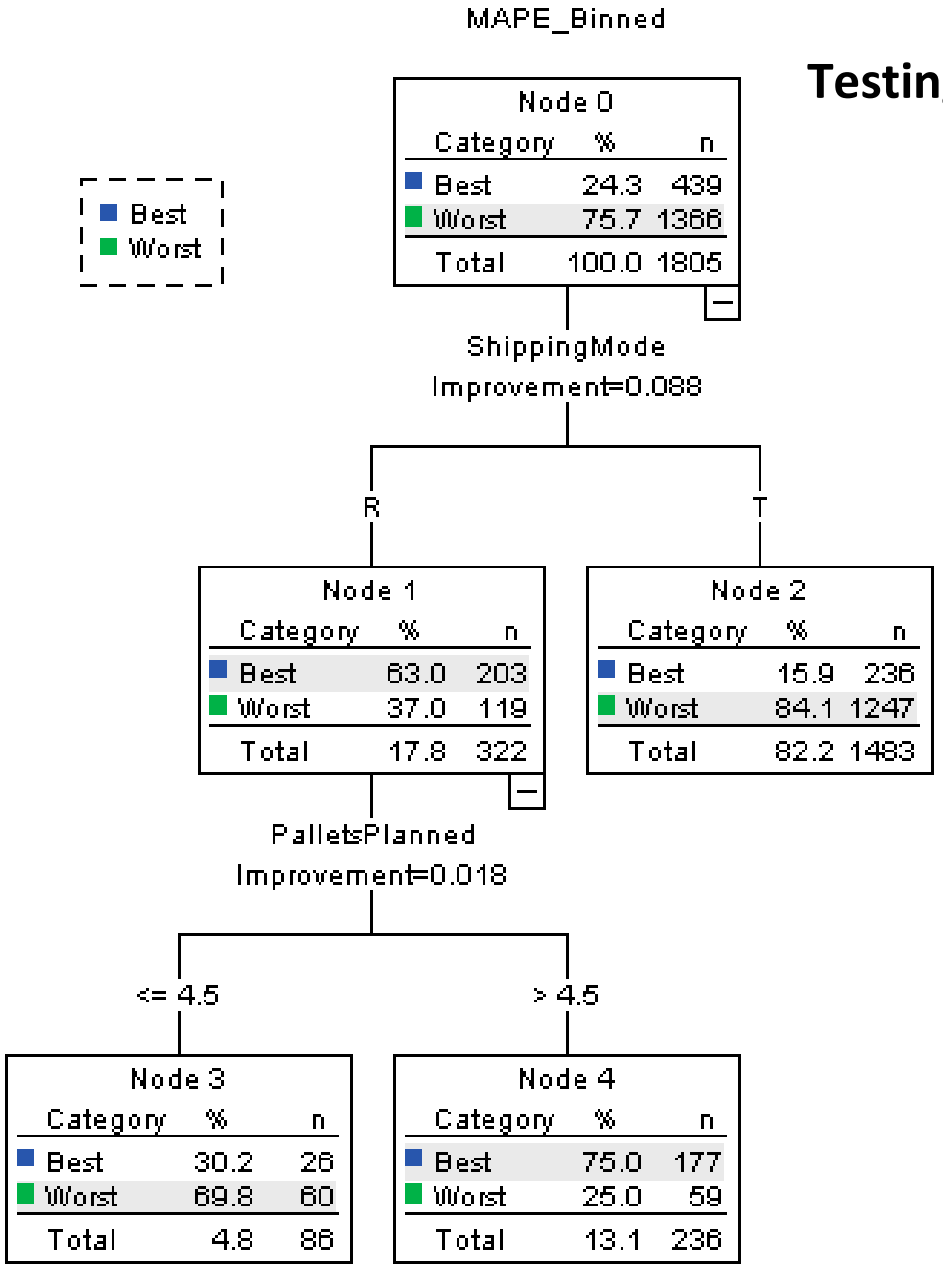
Growing Method: CRT
Dependent Variable: MAPE_Binned



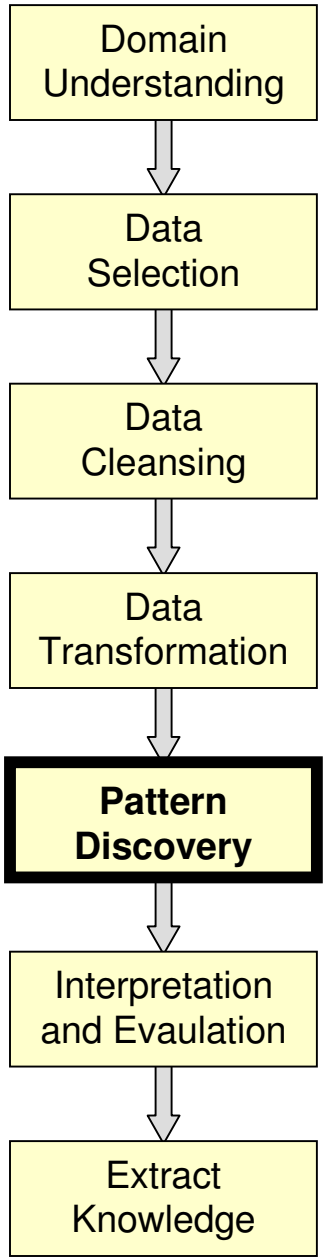


Training (Plan < 50)





Testing (Plan < 50)

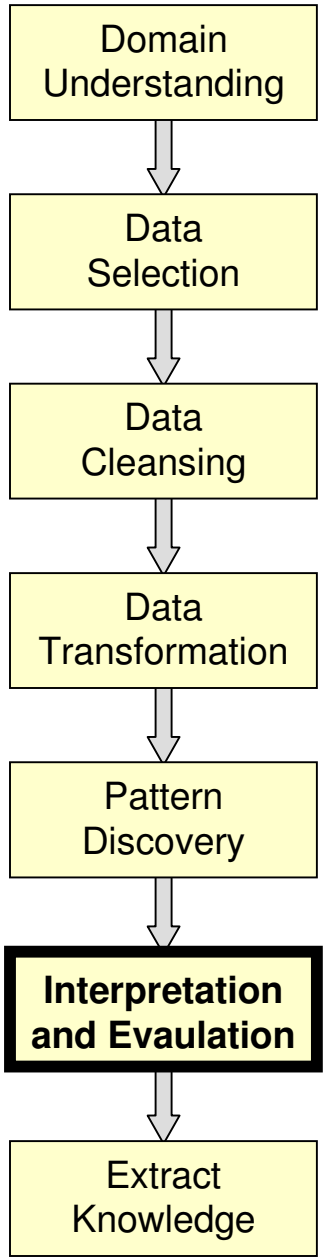


Classification (Plan < 50)

Sample	Observed	Predicted Best	Predicted Worst	Predicted Percent Correct
Training	Best	383	353	52.0%
	Worst	47	1515	97.0%
	Overall Percentage	18.7%	81.3%	82.6%
Test	Best	322	330	49.4%
	Worst	64	1620	96.2%
	Overall Percentage	16.5%	83.5%	83.1%

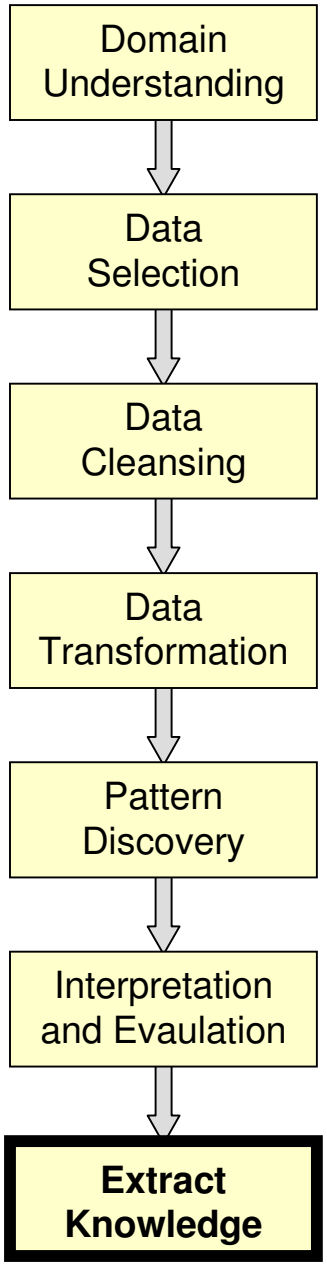
Growing Method: CRT
Dependent Variable: MAPE_Binned

Result was the same for p=100 c=50, p=50 c=25, p=10 c=5, and p=2 c=1.



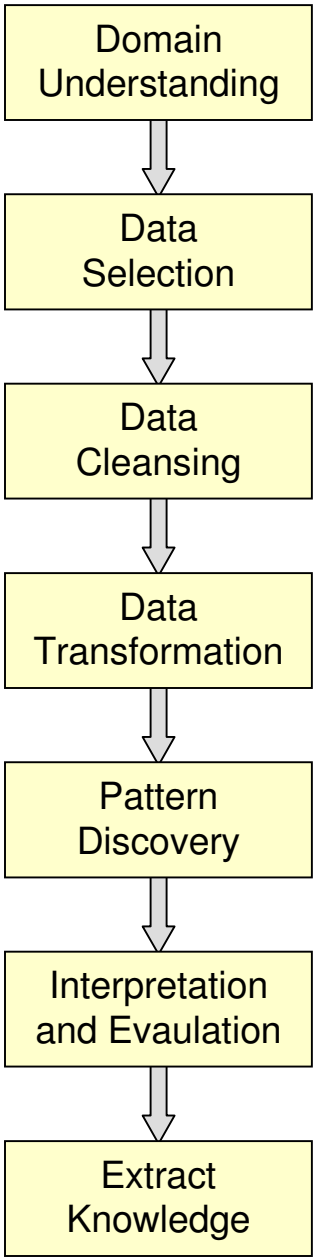
Interpretation and Evaluation

- There were, indeed, very different results for those records with plan >= 50 and those with plan < 50.
- Ultimately, my client will have to determine if the findings are relevent and make business sense. (In particular, I am curious about the cut at 4.5 pallets...?)



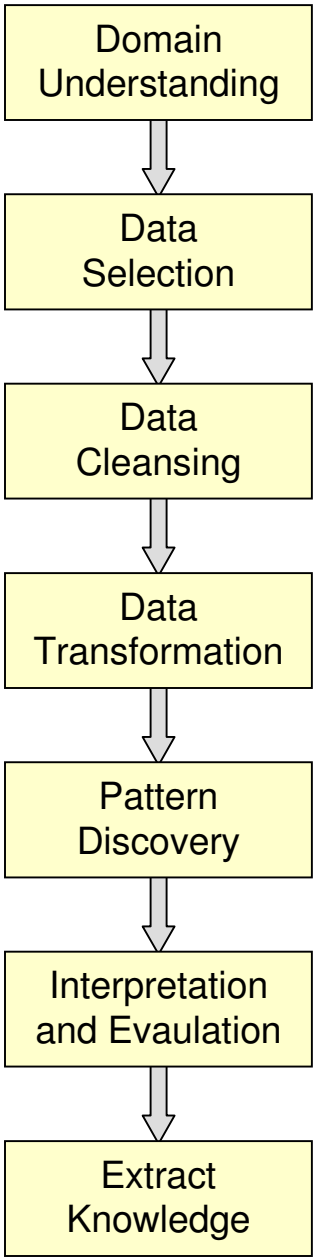
Extract Knowledge

- Knowledge should be
  - non-trivial / non-intuitive
  - previously unknown
  - potentially useful
- My results were rather underwhelming. In fact, I was forewarned that this might be the case.
- Nevertheless, the process shows merit.
- Given the proper variables, I am convinced that the process would isolate the root causes of non-compliance.



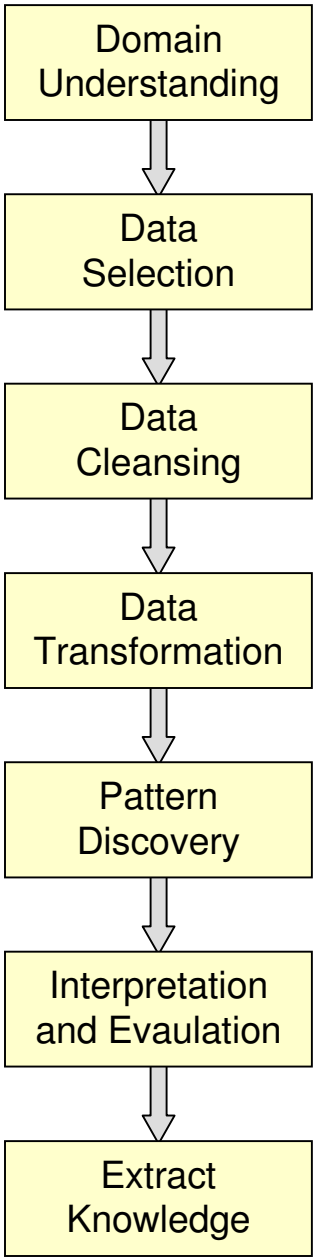
Lessons Learned

- This is harder than it looks! The casual observer will look at the end result and, not seeing all the dead ends I ran into, think it was easy!



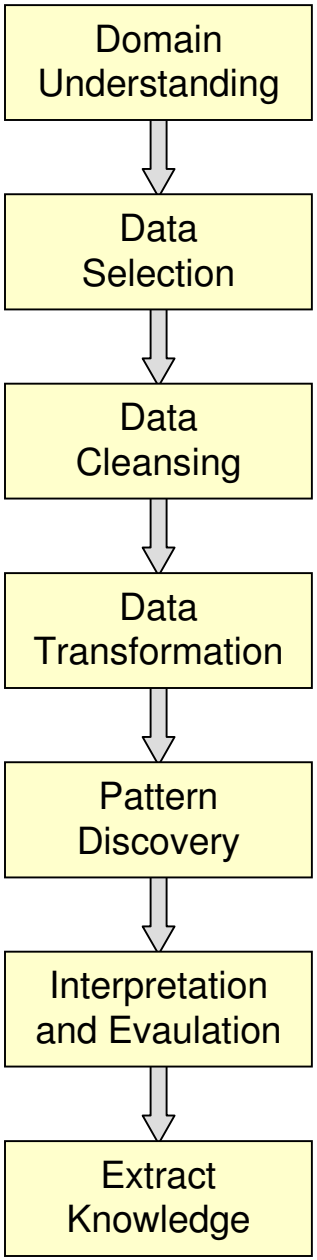
Lessons Learned

- I would prefer a scripting language where I can comment / uncomment my code, making it easier to remember what combinations I have already tried. (This comment may be indicative of my bias for SAS and my current lack of familiarity with SPSS.)



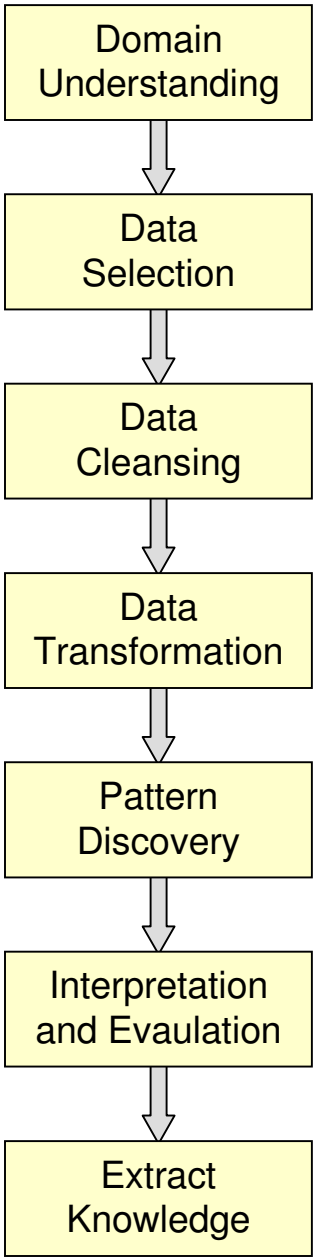
Lessons Learned

- I was really annoyed that SPSS doesn't remember the criteria used for recoding data when you save the dataset. The recoded data is saved, but not the steps you took. I kept Notepad open while I was using SPSS!
- SPSS also does not remember the criteria used in building the decision tree.



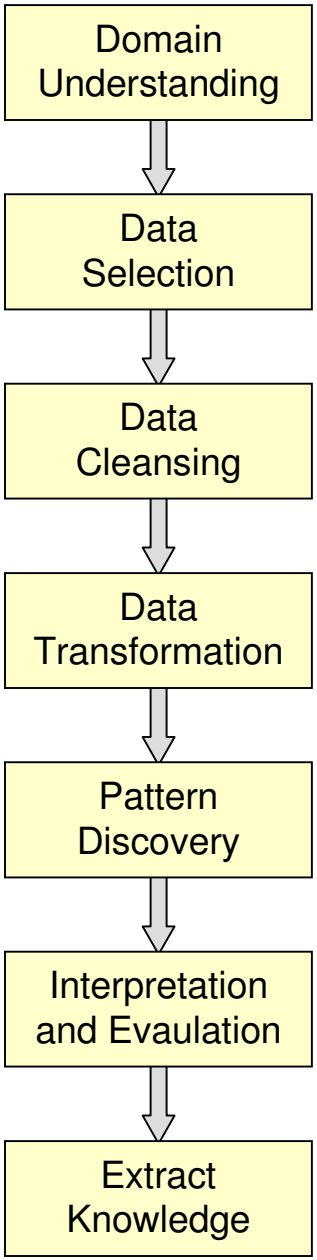
Lessons Learned

- In my first attempts at this, I aggregated the data too early.
- For example, calculating MAPE at the source type, destination type, material group level rather than calculating MAPE at the source, destination, item level and then appending attributes such as source type, destination type, and material group.
- This left me with too few cases.

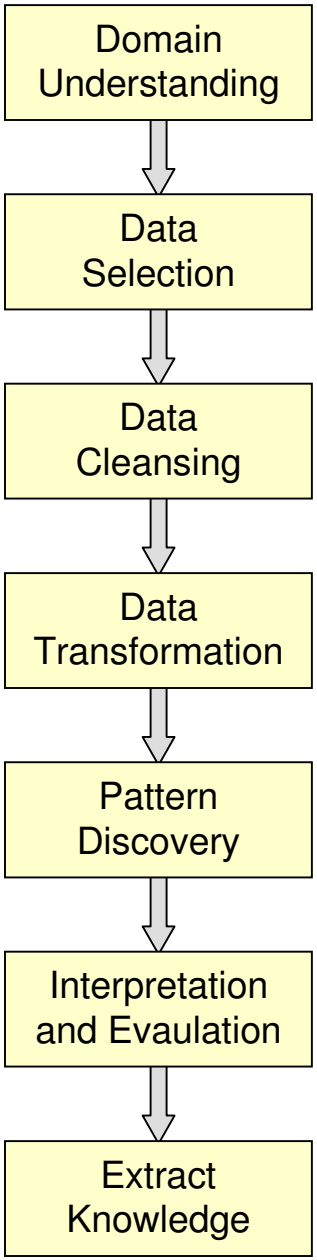


Lessons Learned

- This is SPSS specific: I found it easier to make broad ranges when recoding data, and then using filters to fine tune the inclusion/exclusion of records. For example:
- RECODE MAPE **(0 thru .5=1) (.5 thru Highest=2)** INTO MAPE_Binned.
- COMPUTE filter_$=(PalletsPlanned >= 50 and **((MAPE >= 0 and MAPE <= .2) or MAPE > .8)**).


Domain Understanding
Data Selection
Data Cleansing
Data Transformation
Pattern Discovery
Interpretation and Evaulation
Extract Knowledge

Questions?



References

1. Supply-Chain Analytics: Beyond EPR & SCM. A white paper by IW Custom Research and SAS. Retrieved March 3, 2011 from http://www.sas.com/resources/asset/SAS_IW_FinalLoRes.pdf