DUKE UNIVERSITY
The Fuqua School of Business

MGRECON 491 Data Mining — Gallant
Homework 1 — Term 1
Due September 23, 2008

This homework assignment is both an introduction to the ideas of data mining and a tutorial on the use of XLMiner. For this reason, very detailed instructions are provided.

The Excel workbook bank.xls, which can be downloaded from the course web site, contains data on current and closed loan accounts for a Czech bank. The data are divided into four worksheets: paid, default, current, and arrears, The definitions of the variables whose name appears in the first row of each worksheet are as follows:

district_id	district code for branch location
loan_id	loan identification number
client_id	the average of the client identification numbers of the cosigners
account_id	identifier that links loan account to checking account
date	date of the loan
amount	amount of the loan in Czech currency (Crown $\doteq$ 15\$US)
duration	duration of the loan in months
payments	amount divided by duration
status	coded 1, 2, 3, 4 for paid, defaulted, current, in arrears, respectively
birth_number	the average year of birth of the cosigners
sex	average of the sex id of cosigners where female is coded 1 and male 0
overdraft_p	number of checking account overdraft interest charges prior to the loan period
overdraft_d	number of checking account overdraft interest charges during loan period
numtrans_p	number of transactions prior to the loan period
numtrans_d	number of transactions during the loan period
minbal_p	minimum checking account balance prior to the loan period
minbal_d	minimum checking account balance during the loan period
maxbal_p	maximum checking account balance prior to the loan period
maxbal_d	maximum checking account balance during the loan period
A2	name of district where branch is located
A3	name of region where branch is located
A4	population of district
A5	number of municipalities in district with population less than 499
A6	number of municipalities in district with population within 500–1999
A7	number of municipalities in district with population within 2000–9999
A8	number of municipalities in district with population more than 10000
A9	number of cities in district

A10	percent urban population in district
A11	average salary in district in Czech currency
A12	unemployment rate
A14	number of entrepreneurs per 1000 inhabitants
A15	number of reported crimes in district

Copy the "Paid" and "Default" worksheets to a new worksheet called "Closed." Make sure that the variable names are in the first row of the new worksheet and that they are not duplicated in some other row. If they are duplicated in some other row, delete that row. The worksheet should now have entries in rows 1 through 235 and columns A through AF.

Select the worksheet "Closed." We will now create variable to indicate whether or not a loan defaulted. This information is coded in column I, which is variable status, as 1 for paid and 2 for defaulted but we need it coded as 0 for paid and 1 for defaulted to agree with the conventions of statistical software. To do this, in cell (AG,1) enter the variable name "default." In cell (AG,2) enter the formula "=if(I2=2,1,0)". Drag the formula to the bottom of the column to fill in all rows (or use the double click shortcut). In worksheets "Current" and "Arrears" enter the variable name "default" in cell (AG,1). Do not fill in the remaining rows with data. Save the workbook.

Notice that all the defaults are at the end of worksheet "Closed." Because of this, we need to protect ourselves against a problem. The problem is that there is a tension in data partitioning: One cannot have both an exact 60/40 split between training and validation samples and also have a pure random division without extra programming effort and machine time. Therefore, many programs, and XLMiner appears to be one, do not allocate randomly towards the end of the data. We shall put the data in random order to protect ourselves from this problem. Here is how to put the data in worksheet "Closed" in random order.

In the last empty column of the worksheet "Closed", which should be AH, type "random" in cell (AH,1). In cell (AH,2) enter the formula "=rand()". We need to stop the random numbers from changing every time we do something to the worksheet. This is done as follows. Drag the formula to the bottom to fill in all rows with random numbers. Mark column AH; copy to the clipboard; mark column AI, click on menu item "Edit" above the toolbars, from the pop-up menu select "Paste special," select "Values", click "OK." Delete column AH.

Now we have to sort the entire worksheet on column AH (formerly AI). To do this, we must mark all the data by clicking in the blank cell at the top left of the worksheet, which will mark the entire worksheet. Click on “Data” in the toolbar. Click “Sort.” In the dialog box that pops up make sure “Header row” is checked in the “My list has” panel. Chose “random” in the menu item “Sort by” Click “OK.”

Because of the use of a formula to create “default,” it is a good idea to start afresh with a worksheet that does not depend on entries in other cells so that later deletions do not cause these values to become undefined. To do this open a new worksheet called “Work.” Select all the data in worksheet “Closed” by selecting the worksheet and then clicking on the blank cell at the top left of the worksheet. Click on the copy icon. Select worksheet “Work,” click on menu item “Edit” above the toolbars, as before, from the pop-up menu select “Paste special,” select “Values,” click “OK.” Save the workbook. So we can refer to it later, save it as bank01.xls.

The way that data mining differs from standard small sample statistical inference is reliance on the performance of a model in a validation sample rather than on t-tests, F-tests, etc. So the first thing we have to do is set up our validation sample. Select worksheet “Work.” Click on cell (A,1). Open XLMiner and select “Partition” then “Standard Partition.” The menu “Standard Data Partition” will pop up. In the sub-panel “Data Source” make sure worksheet “Work” is selected. In the sub-panel “Variables,” make sure that “First row contains headers” is selected. Select all variables by dragging the cursor over all the variables and then clicking “ $\geq$.″ The defaults for other choices are “Pick up rows randomly,” “Automatic,” and a 60%–40% split between training and validation samples. Leave these alone because these choices are reasonable for our problem. Click “OK.” A new worksheet “Data_Partition” will be created.

We will now try to gain a feel for what variables will be useful predictors by using the automatic variable selection feature of XLMiner. As explained in class, the best of the automatic data selection criteria available in XLMiner is Mallows Cp.

Select worksheet “Data_Partition.” Open XLMiner. Choose “Prediction” then “Multiple Linear Regression” from the menu. We will cast a wide net to start with. Select the following as input variables: amount, duration, payments, birth_number, sex, overdraft_p, numtrans_p, minbal_p, maxbal_p, A4 though A15. Select default as the output variable.

Note especially that the variable status cannot be used as an input variable because it is the same as the output variable default to within a constant. Be very careful of this in data mining because it is often the case that data contains redundant descriptions of the output variable. The other exclusions were based on common sense or because it would take a ridiculously large number of dummy variable to code them. To be more specific, using loan_id, client_id, etc. as a predictor makes little sense (Barry and Linoff comment on how silly it would be to use them) and overdraft_d, numtrans_d, minbal_d, maxbal_d are not valid predictors for deciding whether or not to approve a loan although they are useful once a loan has been approved, as we shall see below. The variable district_id would require about 100 dummy variables to code and variables A4–A15 would seem to contain the same information as would district_id dummies.

Click "Next." In the dialog box that pops up, click "Best subset" in the "Output options on training data" sub-panel. In the dialog box that pops up, click "Perform best subset selection." Leave "Maximum size of best subset" at 20 and increase "Number of best subsets" to 5. Select "Exhaustive search." Click "OK." Click "Finish." The results we are looking for are in the "Best subset selection" sub-panel of worksheet "MLR_Output1." Sadly, XLMiner does not do a sort for you. You can inspect column Cp in the worksheet "MLR_Output1" for the smallest values or copy and paste special the section of the worksheet labeled "Best subset selection" to a new worksheet and sort on Cp.

The table below is what I found. Construct a similar table for yourself. It will not be exactly the same as mine because the data partitioning is random and your partition will differ from mine. Also, the values of Mallows Cp will be different and may or may not have negative values. All that is important is their order. To illustrate the construction, the two smallest Cp rows of my "Best subset selection" sub-panel were as follows.

#Coeffs	Cp	Model (Constant present in all models)					
6	-2.19015554	Constant	payments	overdraft_p	A8	A10	A15
6	-2.09148335	Constant	payments	overdraft_p	A4	A8	A10

These two rows became the first two columns of the table.

	Mallows C_p											
variable	-2.2	-2.1	-1.8	-1.6	-1.4	-1.2	-1.2	-1.2	-1.0	-1.0	-1.0	-0.9
amount							X	X	X	X	X	X
duration							X	X	X			X
payments	X	X	X	X	X	X						
birth_number												
sex												
overdraft_p	X	X	X	X	X	X	X	X	X	X	X	X
numtrans_p				X	X	X			X			X
minbal_p												
maxbal_p												
A4		X			X		X	X			X	X
A5												
A6												
A7												
A8	X	X	X	X	X	X	X	X	X	X	X	X
A9												
A10	X	X	X	X	X	X	X	X	X	X	X	X
A11			X			X						
A12												
A14												
A15	X			X			X		X	X		

To protect yourself from a crash, save workbook bank01.xls, exit Excel, copy file bank01.xls to bank02.xls. Restart Excel, open bank02.xls and delete all worksheets except "Work," "Closed," "Paid," "Default," "Arrears," and "Data_Partition1."

Click on "XLMiner," "Prediction," "Multiple Linear Regression." In the sub-panel "Data source" of the dialog box "Multiple Linear Regression-Step 1 of 2" that pops up, select "Data_Partition1" as "Worksheet." For "Input variables" in sub-panel "Variables" select the predictors that the table suggests to you would be a good place to start. My selection was payments, overdraft_p, A8, and A10. For "Output variable" select default. Click "Next." In the "Score Training data" and "Score validation data" sub-panels of the dialog box "Multiple Linear Regression-Step 2 of 2" that pops up, check every box. Click the "Finish" button.

You should now find yourself in the "MLR_Output1" worksheet. In "Output Navigator" at the top click "Valid. Score-Detailed Rep." You should jump to worksheet "MLR_ValidScore1." Notice at the top "Back to Navigator." If you click it you will get back to the "Output Navigator" in the "MLR_Output1" worksheet. Clicking on "Navigator" topics is the easiest way to examine MLR output.

We need one other piece of information that XLMiner does not provide: We need to know the classification error rate for the models that we construct. Here is how to compute the classification error rate in the validation sample.

Copy the data in worksheet "MLR_ValidScore1" to the clipboard. Create a new worksheet called "MLR_Rules1" and paste the contents of the clipboard to it using the "Edit," "Paste special," "Values" sequence described above. Delete empty columns. You should be left with variables Row Id, Predicted Value, Actual Value, and Residual as columns A through D. The variable predicted value is the estimated default probability, also called a confidence score. Although it is an estimated probability, it is not guaranteed to be between 0 and 1. Enter the name ErrRule1 in row 1 of the last column. In my case, I have four predictors so it is cell (I,1). For purposes of discussion, I'll assume that yours is the same. Enter the formula =abs(if(b2>0.5,1,0)-c2) in cell (I,2). Drag to the bottom. Every 1 you see in column I is an error made using the rule "Predict default if confidence score larger than 0.5." Compute the average of column I. The easiest way to compute the average is to click the cell below the last entry of column I and use the dialog box gotten by clicking to the right of Σ in the tool bar. That number is your classification error rate using ErrRule1. I got 0.1170. Try another rule, e.g. =abs(if(b2>0.25,1,0)-c2). For this rule I got 0.1489. In my case results were not sensitive to the cut-off rule over a wide range of cut-off values.

Now would be a good time to discuss the regression output. We have seen how one moves about the regression output using the Navigator. You know what most of it means from your previous statistics courses and lecture. If you do not, XLMiner has a good help system that provides the definitions. Most of what has not been discussed in your previous statistics courses or in lecture is irrelevant feature bloat. Lift charts are an exception. Navigate to the training data set lift chart and let us discuss the two charts you see there. Here is the description from the XLMiner help system.

> After the model is built using the training data set, the model is used to score on the training data set and the validation data set (if exists). Then the data set(s) are sorted using the predicted output variable value (or predicted probability of

success in the logistic regression case). After sorting, the actual outcome values of the output variable is cumulated and the lift curve is drawn as number of cases versus the cumulated value. The baseline is drawn as number of cases versus the average of actual output variable values multiplied by the number of cases. The decilewise lift curve is drawn as the decile number versus the cumulative actual output variable value divided by the decile's average output variable value.

What does this mean to us? First let's get the information we need to answer that question. Navigate to "Train. Score-Detailed Rep." What you will see there is a block of data with rows headed "Row Id," "Predicted Value," "Actual Value," "Residual," and "Amount." Copy that block of data to a new worksheet called "LiftData1" using the "Edit," "Paste special," "Values" sequence discussed above. Sort the data "Descending" on "Predicted Value." In my training data set I have 140 cases of which 19 are defaults giving me a default rate of 19/140=0.1357. The 10% of loans that my model predicts are most likely to default are the 14 cases in rows 2 through 15 (row 1 is the header row). In these 14 cases there are 7 actual defaults. Determine what these values are in your training data set.

Now we can answer the question. First consider the straight red line. It tells me how well I would do at predicting defaults if I had no model at all but knew that the average default rate was 0.1357 in which case I would expect that there would be 0.1357x14=1.8998 defaults. My red line plots y=1.8998 at x=14. The entire red line has a slope of 0.1357 with left and right endpoints (x,y)=(0,0) and (x,y)=(140,19), respectively. Determine these values for your red line.

The blue line tells me how well my model does. Of the 14 cases that my model predicts are most likely to default there are 7 actual defaults. Therefore on my blue line I have x=14 and y=7 plotted. (In the block of data below the plots what is shown in decile 1 is 7/14=0.5 for the mean and some other statistics.) Determine these values for your blue line.

My lift at the 1st decile, i.e. at 10% of the cases, which is 14, is 7/1.8998=3.685. This is the value plotted as the first blue bar in the bar chart labeled "Decile-wise lift chart." Lift at the 1st decile is a standard measure of model performance.

In my validation sample I got a lift of 2.6 at the first decile. This suggests possible over-fitting because 3.785 is uncomfortably larger than 2.6. A better way to check is to go back to the Navigator and click "Train. Score-Summary." You should see two blocks of data "Training Data scoring-Summary Report" and "Validation Data scoring - Summary Report." If the "RMS Error" numbers in the training and validation data are about the same, then the model can be expected to generalize well. It would be better if we could compare classification error rates in the training and validation samples but this would involve extra effort to generate a comparison that is very unlikely to lead to a different conclusion than comparing RMSE. However, as we shall see in lecture, RMSE and similar measures computed from the training and validation samples actually estimate different population quantities so that these are just informal checks. Model choice should always be made on the basis of performance in the validation sample.

We will come back and try and build a better regression classifier later. Let us next build a classification tree to see if we learn more about the data first. To protect yourself from a crash, save workbook bank02.xls, exit Excel, copy file bank01.xls to bank03.xls. Restart Excel, open bank03.xls and delete all worksheets except "Work," "Closed," "Paid," "Default," "Arrears," and "Data_Partition1."

Click on XLMiner in the tool bar. Select "Classification" and then "Classification Tree" in the drop down menus. You should have a dialog box that says "Classification Tree-Step 1 of 3" at the top. Make sure that the "Worksheet" in the "Data Source" panel is "Data_Partition1." If you open worksheet "Data_Partition1" before clicking on "XLMiner," you should get the correct worksheet automatically. Select the same input variables as we did for the regression classifier, namely amount, duration, payments, birth_number, sex, overdraft_p, numtrans_p, minbal_p, maxbal_p, A4 though A15. Select default as the output variable. In the "Classes in the output variable" panel you should see "# Classes: 2", "Success class 1," and "cutoff probability value 0.5." Leave these as is. Click on "Next." The dialog box that pops up should have "Normalize input data" unchecked. "Minimum # records in terminal node" checked with the value 14 to the left, and "Prune tree" checked. Leave these as is and click "Next." In the "Trees" panel of dialog box that pops up, put a check in "Full tree," "Best pruned tree," and "Minimum error tree." In the "Score training data" and "Score validation data" panels, check everything. Click "Finish."

Click on "Best Pruned Tree" in the navigator. Mine is a null tree, which is a disappointment. The same for the "Minimum Error Tree." We'll have to see what we can learn from the "Full Tree."

Click on "Full Tree" in the Navigator. My tree cuts first on the variable amount. I suspect that this is the only variable of interest. If you run a regression tree you will find that the average default rate of large loans is about 34%. Your results will differ because of the random partitioning but I doubt that they will differ substantially as to the gross characteristics of the tree.

Click on "Train. Score-Summary" in the Navigator. The error rate is 13.57% because what the tree did with a cut-off of 0.5 is classify every loan as non-default. If we change the cut-off value in the dialog box to 0.3 we get an error rate of 22.86%. Basically all that we have learned from the tree is that large loans are more likely to default. This seems odd. For some reason this bank is not screening its customers well when making large loans. These are consumer loans. By U.S. experience these losses seem large. Look at

https://www.federalreserve.gov/releases/chargeoff/chgallsa.htm

See if you can find equivalent information on East European banks. Also, compute this bank's charge-off rate using the "Closed" worksheet. That is, get the total of all loans and the total of all defaulted loans and express the ratio as a percent.

Let us move on to nearest neighbors. Once again protect yourself by saving book03.xls and start afresh with book04.xls. Because the curse of dimensionality plagues nearest neighbors, we have to be more careful in choosing inputs. Let us use the suggestion from the regression tool and try the variables we used before but due to the tree's suggestion that amount is important we will substitute amount for payment. Our predictors will be amount, overdraft_p, A8, and A10.

Click on XLMiner, "Classification," and "k-nearest neighbors." In the dialog box "k-Nearest Neighbors Classification-Step 1 of 2," make sure the selected worksheet is "Data_Partition1." Select the input variables above and default as the output variable. Click "Next." In the dialog box that pops up choose 5 as "Number of nearest neighbors." Check everything in the "Score training data" and "Score validation data" panels. Click "Finish." Look at the lift charts in the training and validation data sets. Mine are worse that achieved with the regression classifier. Let's repeat with k set to 10. Results are hardly better than no model at all. Let's fiddle with the cutoff value to see if we can do better. This can be done by clicking on "Train. Score-Summary" in the navigator. The classification error rate in the validation sample with a 0.5 cutoff is 12.77%. Changing to 0.4 improves it to 11.7%.

We come now to neural nets. Neural nets are a hard fit and failure to get reasonable results is a common outcome. But occasionally they work spectacularly well and completely dominate other tools. So it's at least worth giving nets a try. Save workbook bank04.xls and start afresh with bank05.xls.

Click "XLMiner," "Classification," "Neural Network." Make sure that the selected worksheet is "Data_Partition1 in the dialog box that pops up. Select inputs amount, duration, payments, birth_number, sex, overdraft_p, numtrans_p, minbal_p, maxbal_p, A4 though A14 and output default. Click "Next." You might as well accept the defaults in the "Step 2 of 3" dialog box. Changing them may help or hurt. It is hard to predict their effect. Trial and error seems to be the only approach. With determination and a lot of tinkering, one might be able to do well. On the other hand, one might just fall into a time sink and emerge with nothing. Click next. In the dialog box that pops up, make sure everything box is checked in the "Score training data" and "Score validation data" panels. Click finish. My results were interesting. There is evidence of over-fitting because the fit in the training sample is much better than in the validation sample. But the net does get good lift in the validation sample. With work, one might be able to get a net that is a real winner.

At this point you have used all the main tools. Summarize your work thus far in a table that gives in one column the name of each tool together with some compact way of describing its variables and tuning parameters (e.g. k for nearest neighbors) and in another column put the classification error rate in the validation sample.

Now build the best classifier that you can for each tool: regression, tree, neural net, k-nearest neighbor. Best means that it generalizes well and has a low classification error rate in the validation sample. Use domain knowledge to generate new variables that may be useful. For example you might try a quadratic term in a numeric variable such as amount squared or income coverage = payments/A11. When adding variables you must partition the data again and you must make sure that the data are partitioned exactly the same as before so that comparisons are valid. Here is how to create a worksheet to which you can add derived variables and that can be partitioned the same.

Save workbook bank05.xls and start afresh with bank06.xls. Open the worksheet "Data_Partition1" and click on "Validation Data" in the "Output Navigator." The data in the panel below will jump to the validation data. On a scrap of paper record the topmost "Row Id." that you see there. In my partition it is 2. Click on "Training Data." That will scroll the data to the first observation. Mark the entire panel of data (i.e., both training and validation) and "Copy," "Paste Special," "Values" to a new worksheet called "DerivedVariables." Delete row 1 and column A if it is blank. Enter "OldRowID." in cell (A,1) without the quotes. In the last column of the worksheet, which is column AJ in my worksheet, enter the header "partition" without the quotes in cell (AJ,1) in my case and probably the same for you. Find the row that has the "Row Id" you wrote on the scrap of paper in the column "OldRowID. In my case it is row 142. Write it down on the scrap. Enter a "t" without the quotes in cell (AJ,2). Drag it to one above the row on the scrap of paper, which is 141 in my case. Enter a "v" without the quotes in the cell below it and drag to the bottom of the data. Now create your derived variables in worksheet "DerivedVariables." Create these exact same derived variables with the exact same names in worksheets "Current" and "Arrears." Be careful here because the variables will not be in the same columns in these worksheets. Now you are ready to partition. Here is how.

Click "XLMiner," "Partition Data," "Standard Partition." In the dialog box that pops up, make sure that "Data source" is worksheet "DerivedVariables." Select all variables in the "Variables" panel except variable partition. In the "Partitioning options" panel check "Use partition variable." Select variable partition by marking partition in the panel above and clicking on the > to the right of "Use partition variable." Click "OK." Your partitioned data will be in worksheet "Data_Partition2." You can check that the partitioning was done correctly by looking at the column with header "OldRowId." The entries should be exactly the same as "Row Id." in worksheet "Data_Partition1."

At this point you have built the best classifier that you can for each tool for the purpose of deciding whether or not to approve a loan. Add all these results to your table of classification error rates in the validation sample. Once you have chosen your best overall tool, which was trained from training sample in worksheet "Data_Partition2," you can go back and retrain it on the full data set in worksheet "DerivedVariables." This improves statistical efficiency, but most data mining practitioners do not bother with this extra step. Our data set is relatively small so that this extra step is probably worth the trouble.

What we want to do now is build the best model for deciding whether a loan that we have already approved is likely to default. For this purpose we are allowed the use of the variables overdraft_d, numtrans_d, minbal_d, and maxbal_d. Build the best classifier that you can using these variables in addition to those that you used above. Use your best classifier to score the worksheets "Current" and "Arrears." Here is how this is done.

Assume, for the sake of discussion, that your best classifier is a regression model using these variables: amount, duration, overdraft_d, A8, A10. Actually, my results are quite good for this model: a lift of 6 in both the validation and training samples.

To score the "Current" data set using these values, sequence through the "Prediction," "Multiple Linear Regression," "Multiple Linear Regression-Step 1 of 2" menus as before. In the "Multiple Linear Regression-Step 2 of 2" menu all else is before except that in the "Score new data" sub-panel you check "In worksheet." A menu "Match variables in the new range" will pop up. In the "Data Source" sub-panel select worksheet "Current." In the "Variables" sub-panel make sure "First row contains headers" is checked and then click on "Match variable(s) with same name(s)." Click "OK." The "Multiple Linear Regression-Step 2 of 2" menu will return. Click "Finish."

Scored data are in worksheet "MLR_NewScore1." To use these data for analysis it is best to "Copy," "Paste Special," "Values" do a new worksheet that I will call "CurrentScored." Sort these data descending on variable Predicted. Those loans at the top of the list are the ones for which default is most likely according to the model.

All tools you consider, be they from the "Prediction" menu or the "Classification" menu will produce a predicted probability of default. This is what you use to rank loans. You are not interested in the 0 or 1 classification that the tool produces for this purpose; you are only interested in the predicted probability. It seems reasonable that the average predicted default probability in the "Arrears" data will be larger than in the "Current" data set if your model is working well. It certainly is for my model. The top of the current list has a score of 0.62; the top of the arrears list has a a score of 3.28. Check this for your model. Prepare a list of the ten most worrisome loans in the "Current" worksheet and the ten most worrisome loans in the "Arrears" worksheet as a demonstration of how your tool can be used to identify problem loans. The most worrisome loans are those for which the predicted default probability is largest.

The best way to organize your thoughts for presenting your work is to think of yourself as a consultant advising this Czech bank. This carries with it the presumption that the technical level of your audience is lower than yours so you will have to explain the ideas behind the tools that you use and your results. Present your results as a PowerPoint presentation that takes no more than twenty minutes to present. Teams will present their work to the class. Here are the main points to address in your presentation.

- Present your results for loan approval.
  - Describe the tool that you used to obtain your results.
  - Describe the entire analysis.
  - Include a lift chart for your tool and explain what information it conveys.
  - Be sure you discuss the notions of loss function and validation.
  - Make sure that your table of classification error loss in the validation sample for each model you fitted is included.
- Present your results for identifying problem loans.
  - Describe the tool that you used to obtain these results. You can skip this if its the same as above.
  - Describe the entire analysis. You can be brief here if similar to above.
  - Include a lift chart for your tool.
  - Make sure that your table of classification error loss in the validation sample for each model you fitted is included.
- Does this banks loss experience seem out of line?

  - Is it at odds with U.S. experience?
  - Is it at odds with Eastern European experience?
  - If it is out of line, make recommendations for improving performance.
    * Be careful here. Do not do something like recommend the purchase of consumer credit information without without making sure that it is available and reliable in the Czech Republic.

What to turn in: Please submit your PowerPoint presentation and your Excel workbooks on a CD. The Excel worksheet files book01.xls, etc. will provide an audit trail where I can trace the results shown in your presentation, if necessary. Your presentation should describe: (a) what you learned about the bank's consumer loan business from your analysis, (b) why you chose the classification model(s) that you did in terms that a layman would understand. Try to follow good principles of statistical presentation i.e., describe the data and try to make its message as clear as possible. Feel free to comment on what you feel might be the underlying causes for the patterns you have observed.

Your presentation should begin with one or two slides that highlight your key conclusions and recommendations especially the bottom-line classification model. Next, it should include some slides that address the background questions addressed. Be sure to include a slide or two describing the data variables and the units in which they were measured. The presentation should include a few slides that illustrate your solution and the process followed in reaching it. You should describe exactly how your tool classifies a loan.