Syllabus on the course “Predictive Modeling”

Approved by Programme Academic Council

Process-verbal 2 from April 10, 2018

|  |  |
| --- | --- |
| Author | Sergey V. Petropavlovsky |
| Credits | 6 |
| Academic Hours | 228 |
| Year of study | 2 |
| Mode of study | Full-time |

1. **Pre-requisites**

Programming (R is a plus but not essential), mathematics (algebra and calculus), probability theory and statistics. Good command of English.

1. **Course Type**

Predictive Modeling is an elective course for second year master students enrolled on the program “Big Data Systems”.

1. **Abstract**

Predictive Modeling is a statistical subject taught to the second year graduate students over the first and second academic modules. The material ranges from classical topics such as linear and non-linear regression and classification to less frequently discussed questions such as Markov Chain Monte-Carlo, hidden Markov chains, dynamic linear models, multivariate time series analysis and co-integration. For each model considered, much attention is paid to performance assessment so as to minimize the forecast error.

Throughout the course a certain balance between mathematical rigor and intuition has to be maintained. Often, this dilemma is resolved in favor of illustrative examples which help students capture the main idea and learn how to use it in practice instead of memorizing derivations. Nonetheless, we find it instructive to provide brief and tractable proofs whenever it makes pedagogical or some other sense. Some not too hard theoretical questions are left for home assignments which makes students work with pen and paper and provides a deeper understanding of underlying theory. The practice skills are developed throughout in-class practice sessions and home assignments involving real-life datasets.

1. **Learning Objectives**

Predictive Modeling gives insight into machine learning algorithms with emphasis on assessing accuracy of prediction and selecting among the models. Another indirect purpose of the course is to guide the students' research by suggesting more challenging topics and problems to the interested students. This kind of activity develops self-study skills and critical thinking, highlights the importance of literature review and many more.

1. **Learning Outcomes**

Upon completion of the course, students are supposed to:

* be aware of understand theory behind predictive modeling, types of predictive models, key steps of model creation and evaluation
* be aware of practical applications of predictive modeling from science to business
* know how to implement different types of models in the R progaramming language
* acquire the skills to use R functions from different R packages to pre-process the input
* apply the knowledge and tools of predictive analytics to real-life applications
1. **Course Plan**

*Topic 1. Predictive modeling process.*

Key parts of predictive models. Concepts of model building. Data ``spending''. Data splitting.

Predictors. Candidate models. Optimal model. Performance estimation. Bias, variance and model complexity. Bias-variance decomposition.

*Topic 2. Overfitting and model tuning*

Concept of overfitting. Model tuning and model evaluation. Tuning parameters. Resampling. k-fold cross-validation. Generalized cross-validation. Bootstrap.

*Topic 3. Linear regression models*

Measuring performance in regression models. Linear regression. Partial least squares and penalized models. The PCA regression.

*Topic 4. Nonlinear regression models*

Neural networks. Multivariate adaptive regression splines. Support vector machines. k-nearest neighbors.

*Topic 5. Linear classification models.*

Measuring performance in classification models. Sensitivity and specificity. Receiver operating characteristic curves. Logistic regression. Linear discriminant analysis. Partial least squares discriminant analysis. Penalized models. Nearest shrunken centroids.

*Topic 6. Nonlinear classification models*

Nonlinear discriminant analysis. Neural networks. Flexible discriminant analysis. Support

vector machines. K-nearest neighbors. Naive Bayes.

*Topic 7. Regression and classification trees.*

Basic regression trees and regression model trees. Basic classification trees. Bagged trees.

Random forests. Boosting.

*Topic 8. Markov Chain Monte Carlo methods*

Goals of Markov Chain Monte Carlo (MCMC). Markov processes. Properties of Markov chains (finiteness, aperiodicity, irreducibility, ergodicity, mixing, etc). The stationary state of the chain. Monte Carlo simulations of distributions. Inverse CDF method. Rejection sampling. The Gibbs sampler. The Metropolis-Hastings algorithm. Issues in chain efficacy. MCMC implementation in R and examples. Applications of MCMC: modeling returns of S&P500 index..

*Topic 9. Dynamic linear models*

Bayesian framework. State space models. Examples of nonlinear and non-Gaussian state space models. State estimation and forecasting: the Kalman filter for dynamic linear models. Smoothing. Controllability and observability of time-invariant DLMs. Filter stability.

*Topic 10. Multivariate linear time series*

Review of univariate analysis of stationary time series. AR(p) time series process. MA(q) time series process. ARMA(p, q) time series process. Multivariate analysis of stationary time series characteristics. Vector autoregressive model. Specification, assumptions and estimation. Diagnostic tests, causality analysis. Forecasting. Structural vector autoregressive model. Specification, assumptions and estimation. Forecast error variance decomposition. Non-stationary time series. Unit root processes. Long-memory processes. Cointegration and common trends. Spurious regression. Concept of cointegration and error-correction models. Systems of cointegrated variables. Granger's representation theorem. Statistical inference for cointegrated systems. Statistical arbitrage. Formation of cointegration pairs. Trading with cointegration pairs.

1. **Reading List**

**Required**

1. Ghatak, Srivastav. Machine Learning with R.– Springer Singapore, 2018. – URL: https://link.springer.com/book/10.1007%2F978-981-10-6808-9 – ЭБС [Springer eBooks (Complete Collection 2017)](http://web.a.ebscohost.com/pfi/ExternalLinkOut/PubFinderLinkOut?sid=d5656562-58c6-46da-8cde-ee1211e73d58@sessionmgr4007&vid=18&Url=http%3a%2f%2fproxylibrary.hse.ru%3a2048%2flogin%3furl%3dhttps%3a%2f%2flink.springer.com%2f10.1007%2f978-981-10-6808-9&Kbid=edp15406453&PackageId=2470716&LinkedFrom=PublicationDetail)
2. Ayyadevara, V. Kishore. Pro Machine Learning Algorithms: A Hands-On Approach to Implementing Algorithms in Python and R.– Apress, 2018. – URL: https://library.books24x7.com/toc.aspx?bookid=142753 – ЭБС [Books 24x7 IT Pro Collection](http://web.a.ebscohost.com/pfi/ExternalLinkOut/PubFinderLinkOut?sid=d5656562-58c6-46da-8cde-ee1211e73d58@sessionmgr4007&vid=20&Url=http%3a%2f%2fproxylibrary.hse.ru%3a2048%2flogin%3furl%3dhttp%3a%2f%2flibrary.books24x7.com%2flibrary.asp%3f%26bookid%3d142753&Kbid=edp18384470&PackageId=1752&LinkedFrom=PublicationDetail)

**Optional**

1. Gupta, Deepti. Applied Analytics Through Case Studies Using SAS and R: Implementing Predictive Models and Machine Learning Techniques.– Apress, 2018. – URL: https://library.books24x7.com/toc.aspx?bookid=143110 – ЭБС [Books 24x7 IT Pro Collection](http://web.a.ebscohost.com/pfi/ExternalLinkOut/PubFinderLinkOut?sid=d5656562-58c6-46da-8cde-ee1211e73d58@sessionmgr4007&vid=20&Url=http%3a%2f%2fproxylibrary.hse.ru%3a2048%2flogin%3furl%3dhttp%3a%2f%2flibrary.books24x7.com%2flibrary.asp%3f%26bookid%3d142753&Kbid=edp18384470&PackageId=1752&LinkedFrom=PublicationDetail)
2. Kubat, Miroslav. An Introduction to Machine Learning– Springer International Publishing, 2017– URL: https://link.springer.com/book/10.1007%2F978-3-319-63913-0 – ЭБС [Springer eBooks (Complete Collection 2017)](http://web.a.ebscohost.com/pfi/ExternalLinkOut/PubFinderLinkOut?sid=d5656562-58c6-46da-8cde-ee1211e73d58@sessionmgr4007&vid=23&Url=http%3a%2f%2fproxylibrary.hse.ru%3a2048%2flogin%3furl%3dhttps%3a%2f%2flink.springer.com%2f10.1007%2f978-3-319-63913-0&Kbid=edp14704283&PackageId=2470716&LinkedFrom=PublicationDetail)
3. **Grading System**

The formula for the final grade 



is made up of the grade  accumulated over the module and the grade  for the final exam. The accumulated grade  is calculated as follows:



where  and  are the grades for the home assignments and the in-class tests, respectively.

1. **Guidelines for Knowledge Assessment**

**Sample concept questions for final exam**

1. Key parts of predictive models. Concepts of model building.

2. Candidate models. Optimal model. Performance estimation.

3. Unsupervised data processing. Techniques of the addition, deletion, transformation of training

data set. Reduction of data skewness or outliers.

4. Feature engineering. Feature extraction.

5. Surrogate variables as combinations of multiple predictors. Dummy variables.

6. Concept of over-fitting.

7. Model tuning and model evaluation. Tuning parameters.

8. Resampling techniques. k-Fold cross-validation. Generalized cross-validation.

9. Bootstrap.

10. Measuring performance in regression models.

11. Linear regression.

12. Partial least squares and penalized models.

13. Neural networks.

14. Multivariate adaptive regression splines.

15. Support vector machines.

16. K-nearest neighbors.

17. Measuring performance in classification models.

18. Sensitivity and specificity. Receiver operating characteristic curves.

19. Logistic regression. Linear discriminant analysis.

20. Partial least squares discriminant analysis.

21. Penalized models.

22. Nearest shrunken centroids.

23. Nonlinear discriminant analysis.

24. Neural networks.

25. Flexible discriminant analysis.

26. Support vector machines.

27. Naive Bayes.

28. Basic classification trees. Bagged trees.

29. Random forests.

30. Boosting. Cubist.

31. Measuring predictor importance. Numeric Outcomes. Categorical Outcomes.

32. Relief algorithm. Model based importance scores.

33. Feature Selections. Non-informative predictors.

34. Markov processes and their properties.

35. Monte Carlo simulations of distributions. Inverse CDF method.

36. The Gibbs sampler.

37. The Metropolis-Hastings algorithm.

38. The Kalman filter.

39. The dynamic linear models.

**Sample practice assignments for final exam**

1. Cross validation. Generate a single dataset described in item 1a. Increase the number of observations to 200. For this dataset, perform a leave-one-out, 5-fold and 10-fold cross validation using the KNN regression. Note that you may have to code the cross validation on your own unless you ﬁnd an appropriate R function. Compare the results with the true test error (item 1d). Select the best model.2. Linear model selection. Choose multivariate data (e.g, built into some package) with not fewer than 10 independent variables. Use command model.matrix to convert qualitative predictors, if any, into dummy variables, see p.248, 251 of [2]. Most predictors should be quantitative.

 (a) Perform the best subset selection to ﬁt linear models with diﬀerent number of predictors on the entire dataset. See Lab 1 wrapping up Chapter 6 of [2].

 (b) Select the best model based on the AIC, BIC, adjusted R2.

 (c) Repeat 2a, 2b for the forward and backward selection. Compare the results with the best subset approach.

 (d) Explain in what sense the models provided by the best subset method 6a and forward/backward 6c are optimal (prior to applying 6b criteria!!!).

 (e) Write a paragraph explaining the derivation of the AIC, formula (7.24) of [1], with emphasis on diﬀerent types of error described by formulae (7.15)-(7.24). No need for reproducing mathematical derivation, just your understanding of the idea.3. Shrinkage and dimension reduction methods for linear models. Use the same data set as for task 6.

 (a) Split the data set into training and test parts randomly. The training and test set should be of approximately the same size.

 (b) Fit the ridge regression, the lasso, the PCR and the PLS on the training set choosing the shrinking parameter λ and number of principal components by cross-validation. Stay on the training set all the time. See Labs 2,3 of [2, Ch.6].

 (c) Compute the test error for all of the models of item 7b. Pick the best model.

 (d) For the ridge regression and lasso, plot the coeﬃcients of the model versus λ (take a discrete array of λ). As an example, see Figures 6.4 and 6.6 of [2].

 (e) Why are the ridge and lasso regression less ﬂexible than the ordinary least squares model?

 (f) The ridge and lasso regression are especially helpful when the sample size ≈ the number of predictors. To check this statement, reduce manually (more preferably, randomly) the number of observations in your data set so that it gets slightly greater than the doubled number of predictors. Split the obtained data set into two equal parts (training and test). Train the ordinary linear model and ridge/lasso regression on the training set and compute the test error. Seek for the optimal shrinkage factor λ, by cross-validation or manually. Compare the errors of the ordinary linear model and ridge/lasso with the optimal λ.

 (g) Principal directions for the PCR. Take the most correlated pair of predictors in your dataset. Use pairs to estimate the correlation visually. Build a matrix X with two columns containing the values of those predictors. Do the singular value decomposition of matrix X = UDVT, see [1, p.64], command svd. Draw a scatter plot of the chosen predictors and impose two lines with the direction vectors v1 and v2 (command abline) where v1 and v2 are the columns of matrix V. Explain what you obtained.

(h) Explain in words the diﬀerence between principal directions in the PCR and the PLS. See [1, p.81].

**10. Methods of Instruction**

In general, lectures should give insight into the concepts and ideas underlying the topic under review. The theoretical core of presentation should be preceded and followed up by clear examples. The lecture slides may contain pieces of code illustrating implementation of the algorithms in some programming language (presumably, in R). It is highly recommended to provide students with the lecture slides prior to the lecture so that they could familiarize themselves with the material in advance and prepare some questions. The lecturer should refer the students for technicalities to the recommended textbooks, reviews and papers as needed throughout the presentation.

Practice classes play the key role in providing the course. The instructor should focus on the implementation of data analysis algorithms on computers. The difficult tasks should be discussed and worked out together with students. The tasks being discussed should be close to those of home assignment so as students could solve similar problems on their own. The students are supposed to prepare a report on a particular home assignment and submit it to the instructor electronically or in paper form. Some requirements for these reports may be set, e.g.:

* The questions should be addressed in the same order they appear in the assignment. The text of the question must be retained and placed before each answer. The working language is English.
* The answer to a particular question may take a form of a plot, formula etc followed by a brief explanation and a conclusion. All conclusions must be justified numerically, i.e., by some computed quantities, plots, etc. The answers do not need to be lengthy but they must be convincing in mathematical and statistical sense, i.e., in terms of some quantitative measures.
* Each student must use a unique data set. It is the student’s responsibility to make sure that no one else is using the same data. To facilitate the distribution of datasets among the students, the instructor can create an editable shared check-in list on Google Drive or some other cloud resource.
* The deadlines for the reports should be clearly specified.
* The instructor should notify the students about the penalties for late submission of the reports.
* The solutions should normally contain code in R or some other language.

It is good practice to suggest the students some datasets for the home assignments. For example, a great amount of market data can be found at Yahoo Finance, Google Finance, Federal Reserve Economic Data repository <http://research.stlouisfed.org/fred2/> and so on. Other possible data sources include the JSE archive http://ww2.amstat.org/publications/jse/jse\_data\_archive.htm, a huge repository at <https://www.data.gov/> and a list of freely available sources at http://guides.emich.edu/data/free-data. Remarkably, most of these data can be downloaded in R directly by using the respective functions which should be pointed out to students.

11. **Special Equipment and Software Support**

|  |  |  |
| --- | --- | --- |
| **#** | **Title** | **Terms of access**  |
| 1. |  Microsoft Windows 7 Professional RUS | Internal HSE network |
| 2. | Microsoft Office Professional Plus 2010 | Internal HSE network |
| 3.  | R programming language | Internal HSE network |
| 4. | R Studio IDE | Internal HSE network |