KU LEUVEN

FACULTEIT INGENIEURSWETENSCHAPPEN

Design, implementation and evaluation of data integration methods for biomedical cancer data

Michiel Ruelens

Thesis voorgedragen tot het behalen van de graad van Master of Science in de ingenieurswetenschappen: computerwetenschappen, hoofdspecialisatie Mens-machine communicatie

Promotor:
Prof. dr. ir. Roel Wuyts & Prof. Olivier Gevaert

Assessoren:
Ir. W. Eetveel
W. Eetrest

Begeleiders:
Ir. A. Assistent
D. Vriend

Academiejaar 2015 – 2016

Preface

I would like to thank everybody who kept me busy the last year, especially my promotor and my assistants. I would also like to thank the jury for reading the text. My sincere gratitude also goes to my wive and the rest of my family.

Michiel Ruelens

Contents

Abstract

The `abstract` environment contains a more extensive overview of the work. But it should be limited to one page.

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget, consectetuer id, vulputate a, magna. Donec vehicula augue eu neque. Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Mauris ut leo. Cras viverra metus rhoncus sem. Nulla et lectus vestibulum urna fringilla ultrices. Phasellus eu tellus sit amet tortor gravida placerat. Integer sapien est, iaculis in, pretium quis, viverra ac, nunc. Praesent eget sem vel leo ultrices bibendum. Aenean faucibus. Morbi dolor nulla, malesuada eu, pulvinar at, mollis ac, nulla. Curabitur auctor semper nulla. Donec varius orci eget risus. Duis nibh mi, congue eu, accumsan eleifend, sagittis quis, diam. Duis eget orci sit amet orci dignissim rutrum.

Samenvatting

In dit `abstract` environment wordt een al dan niet uitgebreide Nederlandse samenvatting van het werk gegeven. Wanneer de tekst voor een Nederlandstalige master in het Engels wordt geschreven, wordt hier normaal een uitgebreide samenvatting verwacht, bijvoorbeeld een tiental bladzijden.

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget, consectetuer id, vulputate a, magna. Donec vehicula augue eu neque. Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Mauris ut leo. Cras viverra metus rhoncus sem. Nulla et lectus vestibulum urna fringilla ultrices. Phasellus eu tellus sit amet tortor gravida placerat. Integer sapien est, iaculis in, pretium quis, viverra ac, nunc. Praesent eget sem vel leo ultrices bibendum. Aenean faucibus. Morbi dolor nulla, malesuada eu, pulvinar at, mollis ac, nulla. Curabitur auctor semper nulla. Donec varius orci eget risus. Duis nibh mi, congue eu, accumsan eleifend, sagittis quis, diam. Duis eget orci sit amet orci dignissim rutrum.

List of Figures and Tables

List of Figures

List of Tables

List of Abbreviations and Symbols

Abbreviations

LoG	Laplacian-of-Gaussian
MSE	Mean Square error
PSNR	Peak Signal-to-Noise ratio

Symbols

42	"The Answer to the Ultimate Question of Life, the Universe, and Everything" according to [**?**]
c	Speed of light
E	Energy
m	Mass
π	The number pi

Chapter 1

Introduction

The first contains a general introduction to the work. The goals are defined and the modus operandi is explained.

1.1 The need for data integration methods

1.2 Goals

1.3 Modus operandi

Chapter 2

Generalized Linear Models

2.1 Introduction

In this chapter I will explain the current standard in machine learning when it comes to generalized linear models. This term indicates a generalization of simple linear regression that allows for a wide range of output variables.
First I will go over the basics of linear models, gradually building up to the definition of generalized linear models. Next, I will describe what actual data looks like and how this data is transformed into a useful model.
After that I will tackle the more recent innovation of regularization that will greatly improve our previous models by exploiting the bias-variance trade-off to reduce overfitting. Lastly I will outline the validation method that will be used to test the performance of the models.

2.2 Classical linear models

When we think of classical linear models, we can imagine a set of numeric explanatory (or input) variables and a numerical dependent (or output) variable. By making a linear combination of the explanatory variables we attempt to estimate a value for the dependent variable. Depending on the type of dependent variable the linear method gets a different name. In the following sections I will outline several of them.

2.2.1 Linear Regression

The simplest version of a linear model is called linear regression. In this case the input variables are combined using a linear combination, and the result of this calculation is immediately used as the final estimate.
//TODO MATH
For the other linear methods we will define a function each time that is applied to the result of the linear combination. We could do the same for linear regression and say that the applied function is the identity function. We could schematize this compuation as follows:

//TODO SCHEMA

2.2.2 Linear Classification

The next method is called linear classification. The difference with linear regression is that we have a different type of output variable. In a classification task we want to predict a class from a list of potential classes. For instance, we could try to predict whether tomorrow will be a sunny day or not. Notice that there are only 2 possible outcomes: 'sunny' or 'not sunny' and we could represent these outcomes as 0 and 1 in our model. This form would be called binary classification because we have 2 possible classes. It is very easy to extend this method to multi-class classification.
The computation in this method starts out exactly the same, combining the input variables using a linear combination. Next, we have to define a threshold to indicate which examples belong to one class or another. In the case of binary classification we would define 1 threshold, and if the result of the linear combination is higher than the threshold we would predict one class. If it is lower, we would predict the other class. The function used here would be called a sign function, which maps real values onto one of 2 possible outcomes. We could represent this computation with the following formula and schema:
//TODO MATH AND SCHEMA

2.2.3 Logistic Regression

The third method I want to present is called logistic regression. In this case, the output variable we want to predict comes from a binomial distribution. This means that they are the result of a probabilistic event. An example would be tossing a coin and checking whether the result is heads or tails. While the outcome is binary (heads or tails) we know that there is an underlying probability for the coin to be heads or tails, and we would like to know this probability.
The idea is still the same. We will make a linear combination of the input variables. However this time we will use a logistic function to produce our estimate. The logistic function is a function that maps real numbers onto the range $[0, 1]$. This result can then be interpreted as an estimate for the probability. We can schematize logistic regression as follows:
//TODO MATH AND SCHEMA

The logistic regression method is the one that will be most widely used throughout this thesis.

2.3 Training a model

In order to understand the integration strategies that will be explained later on, it is useful to know how exactly the models come to be. This section will explain what

the input data for our linear models actually looks like, and how we get from this data to a model that we can use for future predictions.

2.3.1 The data

The data we use consists of two parts: the input data, which can be seen as a matrix where the columns are the explanatory variables and each row is an example (or patient). And secondly the output data, which can be seen as a vector where each value indicates the value of the dependent variable for a single example.
It is easy to see that the length of the output vector has to be equal to the amount of rows in the input matrix, indeed there should be one output value for each example. This amount is often called the size of the dataset and we would like it to be as big as possible. Especially when we are dealing with a large number of explanatory variables, it is essential to have a reasonably amount of examples aswell. This will be discussed in more detail later on //TODO REFERENCE.

2.3.2 Gradient descent

In this section I will explain how we get from the input data to the model. The idea here is that we have some for of error measure. The error measure is a sort of rating for our current model as it indicates how big the mistakes are that our current model is making. There are many different error measures we could use. The one that is used in logistic regression is explained in more detail in the following section.
Once we have a way of computing the error that our current model makes, we can try to minimize this error to obtain our 'best' possible model.

Error measure

In logistic regression the error measure we use is called the cross-entropy error. The formula for this error is the following:

$$E_{in}(w) = \frac{1}{N} \sum_{n=1}^{N} ln(1 + e^{-y_n w^T x_n})$$

where

- x_n is the vector of values for the explanatory variables for example n.
- y_n is the value of the dependent variable for example n.
- w^T is the transpose of the weights vector. These are the parameters of our model that we can adjust.
- N is the size of our dataset.
- $E_{in}(w)$ is the in-sample error. This is the cross-entropy error that we make on the examples in our dataset. It is a function of the weights w.

The origin of this function is explained in appendix //TODO ADD APPENDIX AND REFERENCE. We can however easily notice that this is a reasonable error measure. It is an averaged sum over all examples, where for each example we compute an individual error made on that example.
Notice that $w^T x_n$ is the linear combination of the input variables that our current model suggests. This is the prediction that our current model would make for example n and is a real valued number. On the other hand y_n is the actual correct prediction for example n and has a value of 0 or 1.
If the signs of $w^T x_n$ and y_n agree then our current model actually makes a correct prediction for this example. We can see that in this case the exponential becomes close to 0, making our error for example n very small, as we would expect.
If however their signs are opposite, the exponential becomes larger as our incorrect prediction becomes larger. This in turn will increase the error, again as we would expect.
Thus we can see that if we were to minimize this error, we are moving towards a model that tries to make correct predictions.

The gradient descent method

When trying to minimize a function, a general approach would be to try and compute the derivative of the function, and find the spot where this derivative equals zero. In the case of linear regression it is actually possible to compute this minimum in one step. More details about this can be found in appendix //TODO ADD AND REFERENCE APPENDIX.
In the case of logistic regression however it is not possible to find an analytic solution to this problem. The best we can do is put ourselves somewhere on the error curve and try to move towards the minimum in small steps. This is called an iterative approach.
Remember that our error function looks like this:

$$E_{in}(w) = \frac{1}{N} \sum_{n=1}^{N} ln(1 + e^{-y_n w^T x_n})$$

We can now compute its derivative with respect to w:

$$\nabla E_{in}(w) = -\frac{1}{N} \sum_{n=1}^{N} \frac{y_n x_n}{1 + e^{y_n w^T x_n}}$$

The problem is to find the set of weights w for which the derivative becomes 0 (or that minimizes the error). We can start out with an initial set of weights $w(0)$ and then iteratively update these weights so we move towards the minimum. Let's call the direction in which we update our weights v. The update we make to w then becomes:

$$w(t+1) = w(t) + \eta v$$

where

- $w(t+1)$ are the updated weights for this iteration.
- $w(t)$ are the current weights before we make a move.
- v is a unit vector pointing in the direction we want to move.
- η is a number that indicates how big the move is that we make, also called the step size.

Remember that the gradient of a function at a certain point always points towards the steepest slope upwards. In our case we would like to find the minimum, so it is a good idea to move our weights in the direction of steepest descent. The direction v that we are moving towards then becomes the normalized opposite direction of the gradient:

$$v = -\frac{\nabla E_{in}(w(t))}{\|\nabla E_{in}(w(t))\|}$$

We can now summarize the gradient descent method as follows:

Data: x, y
initialize weights w(0)
while *Stopcondition is not met* **do**
  Compute gradient $\nabla E_{in}(w(t))$
  Compute update direction v
  Update weights $w(t+1) = w(t) + \eta v$
end

Algorithm 1: Gradient Descent algorithm

There are two more non-trivial issues in this computation: the initialization of the weights and the stopcondition.
Weight initialization is sometimes a very tricky thing to do, in the case of logistic regression however it is acceptable to set $w(0)$ equal to the zero-vector as this corresponds to no correlation between any of the input variables and the output variable, and the result of the sigmoid function would be 0.5 or 50% meaning the model has no preference for either outcome.
The stopcondition however is a bigger issue and usually the way to go here is to make a combination of several stop criteria. One criteria would be to simply limit the amount of iterations to a fixed number. This could avoid endlessly overfitting. Another criteria is to set up a target error we want to achieve (a small number), and stop when we have reached this target. This however raises the question of picking the target error, and this is mostly an application dependent choice.
In the version of logistic regression explained here, it can however be shown that the error surface we are dealing with is a very nice convex surface. This makes it very easy to find its minimum and we don't need very complex initialization and stopping criteria to get good results. In other machine learning methods however these surfaces aren't always as nice, and the issue of local minima versus global minima becomes a big deal. There has been much research on this topic however and many sophisticated methods have been developped to deal with this issue.

2.4 Overfitting

Now that we have established a method of computing our models, it is time to deal with an issue known as overfitting. Overfitting points to the fact that there are several mechanisms at work when we are building a model that prevent us from reaching the perfect model (a model that predicts correctly at all times). These mechanisms essentially originate from noise and uncertainty in many aspects of the learning process (the input data, choice of model, choice of algorithm, ...). We can however try to decompose this noise into several components and then attempt to influence them by making changes to our model computation. I will present two ways in which overfitting can be tackled: regularization and validation.

2.4.1 The problem of overfitting

Let's introduce some notation. From now on I will refer to the notion of 'in-sample error' or in symbolic notation E_{in} as the error that a model makes on the examples in our training set. The training set consists of the examples that were used to train (compute) the model in the first place.
Similarly I will define 'out-of-sample error' or E_{out} as the error we make on examples that were not used for training the model. Notice that E_{in} is something we could compute because we have access to the training data, but E_{out} is a quantity we cannot exactly compute but we could try to estimate it if we have some examples left that we did not use for training. Notice also that it is E_{in} that we minimize during our model computation, but it is E_{out} that we actually want to minimize! Indeed, E_{out} corresponds to the error that we get when we are going to deploy our model in practice and use it on examples we have never seen before. We can do this because we believe that E_{in} tracks E_{out} to a certain degree. And thus if we manage to minimize E_{in} we also minimize E_{out} to some extent.
We can only speak of overfitting when we are comparing two models. We say that one model, call it model A, is overfitting with respect to another model, model B, when model A managed to get a lower E_{in} than model B, but model B has a lower E_{out}.
Another way of looking at it is during the learning process. Let's have model A be the model that we computed when we started from model B and performed one more iteration of the training algorithm. Thus model A is 'more trained' than model B. Now let's suppose model A is overfitting:

$$E_{in}^{modelA} < E_{in}^{modelB} \tag{2.1}$$

$$E_{out}^{modelA} > E_{out}^{modelB} \tag{2.2}$$

The additional iteration has decreased the in-sample error, and thus we are able to fit our training data better, but the out-of-sample error has increased, meaning that our model doesn't generalize as well to other examples outside the training set. This means that we are actually fitting our training data too well, while we are not really getting a better grasp of the underlying pattern that we wish to learn. We are

overfitting the training data.

2.4.2 The bias and variance trade-off

There are several ways of looking at overfitting and pointing out its origins. I will introduce the notions of bias and variance and how they can describe the noise in our system.

Average hypothesis

First, let me explain the playing field. We are in a situation of learning, where we are given a set of examples that are produced by some target function f. It is this target function f that we wish to learn (or in other words model). In order to do this we have to decide on the type of functions that we will use to model. In machine learning this is called the hypothesis set. It is the set of all functions that we consider possible candidates to fit our target f. And we will use the examples x in our dataset D to decide which hypothesis we will pick.
Next, let's introduce the notion of average hypothesis $\bar{h}$. Imagine we have a very large number of datasets. For each of these datasets we apply the learning process and we will pick a certain hypothesis h from our hypothesis set. The average hypothesis is then equal to the average of all the hypothesis' just learned. Or in a formula:

$$\bar{h} = \mathbf{E}_D[h^{(D)}]$$

where

- $\bar{h}$ is the average hypothesis.
- $\mathbf{E}_D$ is the expected value over an infinite number of datasets
- $h^{(D)}$ is the hypothesis that was learned for a specific dataset D

We can also look at this average hypothesis as sort of the best we can do with the given hypothesis set. Indeed, when we imagine having an infinite number of datasets we would end up cancelling out much of the variation in the learned hypothesis' and end up with a very good one.

Bias

We can now define the bias as the distance between the average hypothesis $\bar{h}$ and our target function f.

$$bias = (\bar{h} - f)^2$$

We can see the bias as an error we make due to our own choices. Namely our choice of hypothesis set. If we choose a very simple hypothesis set, we cannot expect to

be able to find a fit for a very complex function. The target function simply isn't contained in our hypothesis set. There exists no function in our hypothesis set that exactly fits our target.
Therefore we introduced the notion of average hypothesis. We can view this as the best we can do given our current hypothesis set, and the distance to the target is what we call the bias.

Variance

This however is not the full story. In a real learning situation we generally never find this average hypothesis, because remember it required a large amount (or even infinite amount) of datasets. We never have this luxury! In reality we always have only one dataset and that is all we can use to navigate through the hypothesis set. This is where the notion of variance comes in. We can define variance as the error we get from not having the best hypothesis possible in our hypothesis set. Or in other words as the distance between the hypothesis set that we actually found by learning from our dataset and the average hypothesis.

$$variance = (h^{(D)} - \bar{h})^2$$

Error due to variance mainly comes from two sources: the first is our finite dataset, the second is the complexity of the hypothesis set.
In reality we are given a dataset of N examples and that's all you've got. Most of the time this dataset is not sufficient to find the best hypothesis and thus there will be a variance error made.
Secondly, as we increase the complexity of the hypothesis set, it becomes increasingly difficult to navigate through this set. There are simply many more hypothesis to choose from. Again this makes it harder for our learning process to find the optimal hypothesis and as such will introduce a variance error.

The tradeoff

Having both bias and variance defined we can see that they are not disconnected, there is a tradeoff. If we look purely at bias we could think that simply choosing a super complex hypothesis set is always optimal. Indeed our bias will be zero since the target function will always be inside our hypothesis set.
However, an increasingly complex hypothesis set makes it harder to actually find the optimal hypothesis. We know that the optimal hypothesis is there, but we just cannot find it. The take-away message here is that we have to choose a hypothesis set complexity based on the resources that we have. In this case the resource is our dataset. The larger the dataset that we can learn from, the more complex hypothesis sets we can afford, and the better our results will be. But there is no gain in choosing overly complex hypothesis sets when you don't have the resources to afford them. This will simply cause you to find hypothesis' that fit your training data very well (imagine fitting 3 datapoints with a 7th order polynomial, you would get an exact fit) but this model will not generalize to anything in the real world, it is a complete overfit.

2.5 Regularization

Now that we have the concepts of bias and variance, let's use this information to try and improve our models. The first method is called regularization. In very simple terms this method will add a very small amount of bias in order to greatly decrease the amount of variance, reducing the overall error we make.

2.5.1 Adding bias

Remember that bias is defined as the distance between the average (or best) hypothesis $\bar{h}$ and our target function f. Adding bias effectively means we are going to make another choice, which will impact the average hypothesis. The choice we are about to introduce is based on the following observation: when confronted with a set of similarly performing models, the simplest model is usually the best. Or in other words we should try to prefer simple models over very complex ones.
This observation does not have a mathematical proof, it is rather an observation from experience and reason. One good argument is the fact that noise is usually of high frequency. Meaning that distortions of our dataset (for instance measurement errors) will often be very scattered and random, while the underlying pattern that really makes up the data will be rather smooth. A similar argument can be made for the error due to bias, when we choose a hypothesis set that does not contain the target function, the error due to bias will be mostly random and of high frequency. Therefore if we want to reduce the impact of this noise in our final model, we should prefer models that are not able to fit these high frequencies exactly. Lastly we can remark that if we look at our current understanding of nature (let's say at a larger scale), systems almost always have smooth transitions. The most important laws of nature that we find are all written down in small, simple formulas. Nature doesn't work with instantaneous changes (high frequency), but rather it has smooth functions that govern the basic principle, and then it adds random noise and fluctuations on top of it. This principle is what we try to extrapolate here to machine learning.
Thus, the choice we will make is that we will prefer simple models over complex ones by adding a constraint to the weights.

2.5.2 Regularization types

There are many kinds of constraints that we could add to the weights and, depending on the constraint we choose, the regularization gets a different name and it will have a different effect. One of the most famous regularizers is called ridge or weight-decay. The constraint for this regularizer is the following:

$$\sum_{i=1}^{N} w_i^2 \leq C$$

where

- w_i is the weight (model parameter) for the i'th explanatory variable.

- C is the constraint value

Using this regularizer will result in a preference for models with smaller weights. This keeps certain weights from getting out of control. This form of regularization is also often called the L_2 penalty.
Another popular regularizer is called the lasso penalty (or L_1 penalty). The constraint in this case is:

$$\sum_{i=1}^{N} |w_i| \leq C$$

In addition to keeping the weights small, this form of regularization also performs parameter selection. This means that instead of just keeping the weights small it will also prefer to make weights actually zero. This will cause the resulting model to have fewer parameters, but the parameters that do survive the penalty are sure to be very important. This regularizer is often used when there is a huge number of explanatory variables, and we wish to find only those that are really descriptive. Later in the thesis I will use datasets that contain gene expression information about cancer patients, these datasets often have thousands of explanatory variables and will provide a good example for using the lasso regularization.
The last form of regularizer I wish to demonstrate is called the elastic net penalty. This regularizer is simply a linear combination of the ridge and lasso penalties and provides a way of balancing the two. It has the following constraint:

$$\alpha \sum_{i=1}^{N} w_i^2 + (1-\alpha) \sum_{i=1}^{N} |w_i| \leq C$$

2.5.3 Lambda

Using a regularizer introduces a constraint, this means that we now have to deal with a constrained optimization problem which is much harder to solve than an unconstrained problem. Fortunately, through some clever mathematics it is possible to convert the constrained minimization problem to an unconstrained one by incorporating the regularization constraint in the formula for the error itself. //TODO ADD APPENDIX WITH FULL DERIVATION OR NOT ... The formula for the error with regularization then becomes:

$$E_{in}(w) = \frac{1}{N} \sum_{n=1}^{N} e(x_n, y_n, w) + \lambda R(w)$$

where

- $E_{in}(w)$ is the in-sample error.
- $e(x_n, y_n, w)$ is the individual error made on example n. In the case of logistic regression this would be a cross-entropy error term $ln(1 + e^{-y_n w^T x_n})$.
- λ is the regularization parameter that will be explained below.

- $R(w)$ is the regularization term dependent on the type of regularization used. For instance: $\sum_{i=1}^{N} |w_i|$ for lasso, $\sum_{i=1}^{N} w_i^2$ for ridge, ...
- x_n is the n'th sample in the dataset.
- y_n is the outcome for the n'th sample in the dataset.
- w is the vector of weights, our model parameters that we are trying to find.

Notice that we now again have an error measure that we want to minimize, and it is an unconstrained optimization problem. The important new part is λ. This is the amount of regularization we want to use. It is a new form for the constraint constant C that was earlier introduced in the types of regularization. The higher λ the tighter the constraint (lower C) and vice versa. The value of λ will prove to be critical in getting good models. The way to calculate it is through validation.

2.6 Validation

In this section I will explain the method of validation which is used to estimate the out-of-sample error. I will first explain the issue of sample size and then present a method to work around this limitation.

2.6.1 The sample size dilemma

Remember that when we are training a model we use the in-sample error to navigate the hypothesis space. We do this because we believe that the in-sample error is a valid surrogate for the out-of-sample error (which is the error we actually want to minimize). A valid question to ask is: why don't we just estimate the out-of-sample error and minimize it directly? Consider the following situation:
We are given a dataset of N samples. I will use K samples of the dataset to train my model, this leaves me with $N - K$ samples that are not used for training. Since K samples were used for training, if I would compute the error the model makes on these samples I would be computing the in-sample error. If I want to estimate the out-of-sample error I have to use the $N - K$ samples that were not used for training. This sample set of $N - K$ samples if often called the validation set.
We can now make the following observations:

- The larger we choose K, the more samples are available for training and thus the better our model can be (due to lower variance!)
- The larger we choose K, the less accurate our out-of-sample error estimate is, because we have fewer datapoints for the estimation

So now it is clear that we have a tradeoff to make. We would like K to be as large as possible so that we have a large amount of samples to train a model from. On the other hand we would like K to be as small as possible so that we have enough samples to estimate the out-of-sample error. The solution to this apparent contradiction will be cross-validation.

2.6.2 Cross-validation

Cross-validation is a technique that allows us to have plenty of samples left for training, while still getting a pretty good estimate for the out-of-sample error. The method is as follows: divide the dataset of N samples into K equal parts (often called folds). Each fold now has N/K samples. Train a model on $K-1$ folds and use the remaining fold to estimate the out-of-sample error. Repeat this process K times, one time for each of the K folds, each time leaving out a different fold. In the end we have K estimates of the out-of-sample error and we can average them to get a final result.

Notice that we are really using all samples for validation and training, but never both at the same time. It feels a bit like cheating, but in practice this method works wonderfully. Validation is often used to determine parameters of the learning process, for instance the λ parameter for regularization. We simply try several values for λ, compute the out-of-sample error using validation, and pick the λ that gives the lowest error. Once we have decided this λ we can then train a model on the full dataset of N points and use the λ we have just calculated to be optimal.

We have to remark however that when we use validation to calculate a value for λ as described above, we are really using the validation to help train the model. In this case we can no longer make the statement that the validation samples are not used for training. This is called data pollution. We are using the same data to train training parameters aswell as training the model itself and it is obvious that this will give rise to additional correlations in the data. However in practice it is generally accepted that if you use this technique to decide on just a few learning parameters (often just λ), and you have a big enough dataset, the data pollution is minimal and the results and estimates you get are still reliable.

Cross-validation is not only used to estimate learning parameters. It can also be used to simply test the performance of the model. In this case the fold that is not used for training is usually not called a validation set, but rather a test set. Because samples in this set will be used to test the models performance.

2.7 Conclusion

We have now covered the basis of Generalized Linear Models. We have described the gradient descent method that we can use to navigate the hypothesis space by minimizing an error function. We have seen that overfitting is a serious issue that is caused by noise at different levels of the learning process. We have broken down this noise into bias and variance and shown that we can have an impact on this process. We can use regularization to add a slight bias in order to greatly decrease the error due to variance and we have based this on the principle that we should prefer simple and smooth models. Lastly we have covered the method of validation. A method

that we can use to estimate the out-of-sample error and which gives us the ability to choose learning parameters like λ and also test the performance of our model.

Chapter 3

Integration Strategies

Chapter 4

Evaluation of integration strategies

4.1 Introduction

4.2 Predicting stage outcome

4.3 Predicting survival curves

4.4 Conclusion

Chapter 5

Tool for automated evaluation

5.1 Introduction

5.2 Technologies

5.3 Demonstration

5.4 Conclusion

Chapter 6

Conclusion

The final chapter contains the overall conclusion. It also contains suggestions for future work and industrial applications.

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget, consectetuer id, vulputate a, magna. Donec vehicula augue eu neque. Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Mauris ut leo. Cras viverra metus rhoncus sem. Nulla et lectus vestibulum urna fringilla ultrices. Phasellus eu tellus sit amet tortor gravida placerat. Integer sapien est, iaculis in, pretium quis, viverra ac, nunc. Praesent eget sem vel leo ultrices bibendum. Aenean faucibus. Morbi dolor nulla, malesuada eu, pulvinar at, mollis ac, nulla. Curabitur auctor semper nulla. Donec varius orci eget risus. Duis nibh mi, congue eu, accumsan eleifend, sagittis quis, diam. Duis eget orci sit amet orci dignissim rutrum.

Nam dui ligula, fringilla a, euismod sodales, sollicitudin vel, wisi. Morbi auctor lorem non justo. Nam lacus libero, pretium at, lobortis vitae, ultricies et, tellus. Donec aliquet, tortor sed accumsan bibendum, erat ligula aliquet magna, vitae ornare odio metus a mi. Morbi ac orci et nisl hendrerit mollis. Suspendisse ut massa. Cras nec ante. Pellentesque a nulla. Cum sociis natoque penatibus et magnis dis parturient montes, nascetur ridiculus mus. Aliquam tincidunt urna. Nulla ullamcorper vestibulum turpis. Pellentesque cursus luctus mauris.

Nulla malesuada porttitor diam. Donec felis erat, congue non, volutpat at, tincidunt tristique, libero. Vivamus viverra fermentum felis. Donec nonummy pellentesque ante. Phasellus adipiscing semper elit. Proin fermentum massa ac quam. Sed diam turpis, molestie vitae, placerat a, molestie nec, leo. Maecenas lacinia. Nam ipsum ligula, eleifend at, accumsan nec, suscipit a, ipsum. Morbi blandit ligula feugiat magna. Nunc eleifend consequat lorem. Sed lacinia nulla vitae enim. Pellentesque tincidunt purus vel magna. Integer non enim. Praesent euismod nunc eu purus. Donec bibendum quam in tellus. Nullam cursus pulvinar lectus. Donec et mi. Nam vulputate metus eu enim. Vestibulum pellentesque felis eu massa.

Quisque ullamcorper placerat ipsum. Cras nibh. Morbi vel justo vitae lacus tincidunt ultrices. Lorem ipsum dolor sit amet, consectetuer adipiscing elit. In

hac habitasse platea dictumst. Integer tempus convallis augue. Etiam facilisis. Nunc elementum fermentum wisi. Aenean placerat. Ut imperdiet, enim sed gravida sollicitudin, felis odio placerat quam, ac pulvinar elit purus eget enim. Nunc vitae tortor. Proin tempus nibh sit amet nisl. Vivamus quis tortor vitae risus porta vehicula.

Fusce mauris. Vestibulum luctus nibh at lectus. Sed bibendum, nulla a faucibus semper, leo velit ultricies tellus, ac venenatis arcu wisi vel nisl. Vestibulum diam. Aliquam pellentesque, augue quis sagittis posuere, turpis lacus congue quam, in hendrerit risus eros eget felis. Maecenas eget erat in sapien mattis porttitor. Vestibulum porttitor. Nulla facilisi. Sed a turpis eu lacus commodo facilisis. Morbi fringilla, wisi in dignissim interdum, justo lectus sagittis dui, et vehicula libero dui cursus dui. Mauris tempor ligula sed lacus. Duis cursus enim ut augue. Cras ac magna. Cras nulla. Nulla egestas. Curabitur a leo. Quisque egestas wisi eget nunc. Nam feugiat lacus vel est. Curabitur consectetuer.

Suspendisse vel felis. Ut lorem lorem, interdum eu, tincidunt sit amet, laoreet vitae, arcu. Aenean faucibus pede eu ante. Praesent enim elit, rutrum at, molestie non, nonummy vel, nisl. Ut lectus eros, malesuada sit amet, fermentum eu, sodales cursus, magna. Donec eu purus. Quisque vehicula, urna sed ultricies auctor, pede lorem egestas dui, et convallis elit erat sed nulla. Donec luctus. Curabitur et nunc. Aliquam dolor odio, commodo pretium, ultricies non, pharetra in, velit. Integer arcu est, nonummy in, fermentum faucibus, egestas vel, odio.

Sed commodo posuere pede. Mauris ut est. Ut quis purus. Sed ac odio. Sed vehicula hendrerit sem. Duis non odio. Morbi ut dui. Sed accumsan risus eget odio. In hac habitasse platea dictumst. Pellentesque non elit. Fusce sed justo eu urna porta tincidunt. Mauris felis odio, sollicitudin sed, volutpat a, ornare ac, erat. Morbi quis dolor. Donec pellentesque, erat ac sagittis semper, nunc dui lobortis purus, quis congue purus metus ultricies tellus. Proin et quam. Class aptent taciti sociosqu ad litora torquent per conubia nostra, per inceptos hymenaeos. Praesent sapien turpis, fermentum vel, eleifend faucibus, vehicula eu, lacus.

Appendices

Appendix A

The First Appendix

Appendices hold useful data which is not essential to understand the work done in the master thesis. An example is a (program) source. An appendix can also have sections as well as figures and references[**?**].

A.1 More Lorem

Quisque facilisis auctor sapien. Pellentesque gravida hendrerit lectus. Mauris rutrum sodales sapien. Fusce hendrerit sem vel lorem. Integer pellentesque massa vel augue. Integer elit tortor, feugiat quis, sagittis et, ornare non, lacus. Vestibulum posuere pellentesque eros. Quisque venenatis ipsum dictum nulla. Aliquam quis quam non metus eleifend interdum. Nam eget sapien ac mauris malesuada adipiscing. Etiam eleifend neque sed quam. Nulla facilisi. Proin a ligula. Sed id dui eu nibh egestas tincidunt. Suspendisse arcu.

A.1.1 Lorem 15–17

Nulla in ipsum. Praesent eros nulla, congue vitae, euismod ut, commodo a, wisi. Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Aenean nonummy magna non leo. Sed felis erat, ullamcorper in, dictum non, ultricies ut, lectus. Proin vel arcu a odio lobortis euismod. Vestibulum ante ipsum primis in faucibus orci luctus et ultrices posuere cubilia Curae; Proin ut est. Aliquam odio. Pellentesque massa turpis, cursus eu, euismod nec, tempor congue, nulla. Duis viverra gravida mauris. Cras tincidunt. Curabitur eros ligula, varius ut, pulvinar in, cursus faucibus, augue.

Nulla mattis luctus nulla. Duis commodo velit at leo. Aliquam vulputate magna et leo. Nam vestibulum ullamcorper leo. Vestibulum condimentum rutrum mauris. Donec id mauris. Morbi molestie justo et pede. Vivamus eget turpis sed nisl cursus tempor. Curabitur mollis sapien condimentum nunc. In wisi nisl, malesuada at, dignissim sit amet, lobortis in, odio. Aenean consequat arcu a ante. Pellentesque porta elit sit amet orci. Etiam at turpis nec elit ultricies imperdiet. Nulla facilisi.

In hac habitasse platea dictumst. Suspendisse viverra aliquam risus. Nullam pede justo, molestie nonummy, scelerisque eu, facilisis vel, arcu.

Curabitur tellus magna, porttitor a, commodo a, commodo in, tortor. Donec interdum. Praesent scelerisque. Maecenas posuere sodales odio. Vivamus metus lacus, varius quis, imperdiet quis, rhoncus a, turpis. Etiam ligula arcu, elementum a, venenatis quis, sollicitudin sed, metus. Donec nunc pede, tincidunt in, venenatis vitae, faucibus vel, nibh. Pellentesque wisi. Nullam malesuada. Morbi ut tellus ut pede tincidunt porta. Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Etiam congue neque id dolor.

A.1.2 Lorem 18–19

Donec et nisl at wisi luctus bibendum. Nam interdum tellus ac libero. Sed sem justo, laoreet vitae, fringilla at, adipiscing ut, nibh. Maecenas non sem quis tortor eleifend fermentum. Etiam id tortor ac mauris porta vulputate. Integer porta neque vitae massa. Maecenas tempus libero a libero posuere dictum. Vestibulum ante ipsum primis in faucibus orci luctus et ultrices posuere cubilia Curae; Aenean quis mauris sed elit commodo placerat. Class aptent taciti sociosqu ad litora torquent per conubia nostra, per inceptos hymenaeos. Vivamus rhoncus tincidunt libero. Etiam elementum pretium justo. Vivamus est. Morbi a tellus eget pede tristique commodo. Nulla nisl. Vestibulum sed nisl eu sapien cursus rutrum.

Nulla non mauris vitae wisi posuere convallis. Sed eu nulla nec eros scelerisque pharetra. Nullam varius. Etiam dignissim elementum metus. Vestibulum faucibus, metus sit amet mattis rhoncus, sapien dui laoreet odio, nec ultricies nibh augue a enim. Fusce in ligula. Quisque at magna et nulla commodo consequat. Proin accumsan imperdiet sem. Nunc porta. Donec feugiat mi at justo. Phasellus facilisis ipsum quis ante. In ac elit eget ipsum pharetra faucibus. Maecenas viverra nulla in massa.

A.2 Lorem 51

Maecenas dui. Aliquam volutpat auctor lorem. Cras placerat est vitae lectus. Curabitur massa lectus, rutrum euismod, dignissim ut, dapibus a, odio. Ut eros erat, vulputate ut, interdum non, porta eu, erat. Cras fermentum, felis in porta congue, velit leo facilisis odio, vitae consectetuer lorem quam vitae orci. Sed ultrices, pede eu placerat auctor, ante ligula rutrum tellus, vel posuere nibh lacus nec nibh. Maecenas laoreet dolor at enim. Donec molestie dolor nec metus. Vestibulum libero. Sed quis erat. Sed tristique. Duis pede leo, fermentum quis, consectetuer eget, vulputate sit amet, erat.

Appendix B

The Last Appendix

Appendices are numbered with letters, but the sections and subsections use arabic numerals, as can be seen below.

B.1 Lorem 20-24

Nulla ac nisl. Nullam urna nulla, ullamcorper in, interdum sit amet, gravida ut, risus. Aenean ac enim. In luctus. Phasellus eu quam vitae turpis viverra pellentesque. Duis feugiat felis ut enim. Phasellus pharetra, sem id porttitor sodales, magna nunc aliquet nibh, nec blandit nisl mauris at pede. Suspendisse risus risus, lobortis eget, semper at, imperdiet sit amet, quam. Quisque scelerisque dapibus nibh. Nam enim. Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Nunc ut metus. Ut metus justo, auctor at, ultrices eu, sagittis ut, purus. Aliquam aliquam.

Etiam pede massa, dapibus vitae, rhoncus in, placerat posuere, odio. Vestibulum luctus commodo lacus. Morbi lacus dui, tempor sed, euismod eget, condimentum at, tortor. Phasellus aliquet odio ac lacus tempor faucibus. Praesent sed sem. Praesent iaculis. Cras rhoncus tellus sed justo ullamcorper sagittis. Donec quis orci. Sed ut tortor quis tellus euismod tincidunt. Suspendisse congue nisl eu elit. Aliquam tortor diam, tempus id, tristique eget, sodales vel, nulla. Praesent tellus mi, condimentum sed, viverra at, consectetuer quis, lectus. In auctor vehicula orci. Sed pede sapien, euismod in, suscipit in, pharetra placerat, metus. Vivamus commodo dui non odio. Donec et felis.

Etiam suscipit aliquam arcu. Aliquam sit amet est ac purus bibendum congue. Sed in eros. Morbi non orci. Pellentesque mattis lacinia elit. Fusce molestie velit in ligula. Nullam et orci vitae nibh vulputate auctor. Aliquam eget purus. Nulla auctor wisi sed ipsum. Morbi porttitor tellus ac enim. Fusce ornare. Proin ipsum enim, tincidunt in, ornare venenatis, molestie a, augue. Donec vel pede in lacus sagittis porta. Sed hendrerit ipsum quis nisl. Suspendisse quis massa ac nibh pretium cursus. Sed sodales. Nam eu neque quis pede dignissim ornare. Maecenas eu purus ac urna tincidunt congue.

Donec et nisl id sapien blandit mattis. Aenean dictum odio sit amet risus. Morbi purus. Nulla a est sit amet purus venenatis iaculis. Vivamus viverra purus vel

magna. Donec in justo sed odio malesuada dapibus. Nunc ultrices aliquam nunc. Vivamus facilisis pellentesque velit. Nulla nunc velit, vulputate dapibus, vulputate id, mattis ac, justo. Nam mattis elit dapibus purus. Quisque enim risus, congue non, elementum ut, mattis quis, sem. Quisque elit.

Maecenas non massa. Vestibulum pharetra nulla at lorem. Duis quis quam id lacus dapibus interdum. Nulla lorem. Donec ut ante quis dolor bibendum condimentum. Etiam egestas tortor vitae lacus. Praesent cursus. Mauris bibendum pede at elit. Morbi et felis a lectus interdum facilisis. Sed suscipit gravida turpis. Nulla at lectus. Vestibulum ante ipsum primis in faucibus orci luctus et ultrices posuere cubilia Curae; Praesent nonummy luctus nibh. Proin turpis nunc, congue eu, egestas ut, fringilla at, tellus. In hac habitasse platea dictumst.

B.2 Lorem 25-27

Vivamus eu tellus sed tellus consequat suscipit. Nam orci orci, malesuada id, gravida nec, ultricies vitae, erat. Donec risus turpis, luctus sit amet, interdum quis, porta sed, ipsum. Suspendisse condimentum, tortor at egestas posuere, neque metus tempor orci, et tincidunt urna nunc a purus. Sed facilisis blandit tellus. Nunc risus sem, suscipit nec, eleifend quis, cursus quis, libero. Curabitur et dolor. Sed vitae sem. Cum sociis natoque penatibus et magnis dis parturient montes, nascetur ridiculus mus. Maecenas ante. Duis ullamcorper enim. Donec tristique enim eu leo. Nullam molestie elit eu dolor. Nullam bibendum, turpis vitae tristique gravida, quam sapien tempor lectus, quis pretium tellus purus ac quam. Nulla facilisi.

Duis aliquet dui in est. Donec eget est. Nunc lectus odio, varius at, fermentum in, accumsan non, enim. Aliquam erat volutpat. Proin sit amet nulla ut eros consectetuer cursus. Phasellus dapibus aliquam justo. Nunc laoreet. Donec consequat placerat magna. Duis pretium tincidunt justo. Sed sollicitudin vestibulum quam. Nam quis ligula. Vivamus at metus. Etiam imperdiet imperdiet pede. Aenean turpis. Fusce augue velit, scelerisque sollicitudin, dictum vitae, tempor et, pede. Donec wisi sapien, feugiat in, fermentum ut, sollicitudin adipiscing, metus.

Donec vel nibh ut felis consectetuer laoreet. Donec pede. Sed id quam id wisi laoreet suscipit. Nulla lectus dolor, aliquam ac, fringilla eget, mollis ut, orci. In pellentesque justo in ligula. Maecenas turpis. Donec eleifend leo at felis tincidunt consequat. Aenean turpis metus, malesuada sed, condimentum sit amet, auctor a, wisi. Pellentesque sapien elit, bibendum ac, posuere et, congue eu, felis. Vestibulum mattis libero quis metus scelerisque ultrices. Sed purus.

Bibliography

Fiche masterproef

Student: Michiel Ruelens

Titel: Design, implementation and evaluation of data integration methods for biomedical cancer data

Nederlandse titel: Design, implementatie en evaluatie van data integratie methoden voor biomedische data

UDC: 621.3

Korte inhoud:

Hier komt een heel bondig abstract van hooguit 500 woorden. LaTeX commando's mogen hier gebruikt worden. Blanco lijnen (of het commando `\par`) zijn wel niet toegelaten!

Nam dui ligula, fringilla a, euismod sodales, sollicitudin vel, wisi. Morbi auctor lorem non justo. Nam lacus libero, pretium at, lobortis vitae, ultricies et, tellus. Donec aliquet, tortor sed accumsan bibendum, erat ligula aliquet magna, vitae ornare odio metus a mi. Morbi ac orci et nisl hendrerit mollis. Suspendisse ut massa. Cras nec ante. Pellentesque a nulla. Cum sociis natoque penatibus et magnis dis parturient montes, nascetur ridiculus mus. Aliquam tincidunt urna. Nulla ullamcorper vestibulum turpis. Pellentesque cursus luctus mauris.

Thesis voorgedragen tot het behalen van de graad van Master of Science in de ingenieurswetenschappen: computerwetenschappen, hoofdspecialisatie Mens-machine communicatie

Promotor: Prof. dr. ir. Roel Wuyts & Prof. Olivier Gevaert

Assessoren: Ir. W. Eetveel
W. Eetrest

Begeleiders: Ir. A. Assistent
D. Vriend