MIA_example

Membership Inference Attacks

In this notebook, we describe some of the potential consequences if we fail to ensure security on TRE releases.

We will look at a particular problem which can arise *without* malicious actions by researchers or TRE staff. In other words, our scenario is one which can happen even if all researchers and TRE staff are well-intentioned.

In order to illustrate the point in a setting which makes sense, we will use only a simulated data set: all `patients`, `samples` and so on are fictional.

We will consider a scenario in which we recruit a group of drug users, with the intent of trying to predict bad financial outcomes in order to help other drug users avoid these. We presume that we recruit a range of people with the assurance that their participation is anonymous. We will show that we may inadvertently give away the details of the people who participated, even without directly releasing their data.

This example is extreme; technically, it uses a very *overfitted* model. This is largel just to make the effects obvious; the same effects can occur in less extreme settings.

Throughout this document, text and output in red will indicate *private data* which is protected on a TRE. Text and output in blue will indicate data that is publically available. We will show that using only publically available data we can work out some data that was only privately available.

Problem overview

Intravenous (IV) drug users in the community often face financial difficulties, due to a range of factors. These financial difficulties can make it harder to address drug related problems and have serious effects on quality of life. In order to help IV drug users avoid severe financial difficulties, we are interested in answering the following research question:

> Given a particular IV drug user, what is the probability they are financially insolvent?

In order to do this we recruit 50 IV drug users *with the promise that their participation in the study will be anonymous*. We record whether they are financially solvent, along with their age, sex, level of education, housing status, number of previous periods of rehabilitation, and number of previous drug convictions.

Here are the first ten rows of the our private data:

```
head(data_matrix)
```

```
##      names age      education sex prev_convictions   housing prev_rehab solvent
## 1 Michelle  34 Some_secondary   F                2     Other          0       1
## 2  Lindsay  35      Secondary   M                0 Supported          1       1
## 3    James  40     University   F                2     Other          4       1
## 4  Kenneth  36     University   M                3     Other          1       1
## 5    Emily  32           None   M                1  No_fixed          2       0
## 6  Rebecca  22 Some_secondary   M                0 Supported          0       0
```

Importantly, the status of these individuals as being *included in the study* is private:

```
head(in_study)
```

```
##       name age in_study
## 1 Michelle  34 In study
```

```
## 2  Lindsay  35 In study
## 3    James  40 In study
## 4  Kenneth  36 In study
## 5    Emily  32 In study
## 6  Rebecca  22 In study
```

We plan to learn a rule to predict the variable `solvent` on the basis of the other variables. The details will not matter much, but in this case we will use technical tool called a Gaussian Process classifier.

Data analysis (on safe haven)

We now try and learn our rule from the data. It will turn out that the rule we learn is not very good (it is 'overfitted', meaning that it works well on the data we already have, but would not work well if we were to try and use it on new people).

```
mod1=gausspr(solvent~.,data=data_matrix[,-1],kpar=list(sigma=3))
```

We release this rule (in the form of a model) to the public (now we are blue):

```
summary(mod1)
```

```
##  Length   Class    Mode
##       1 gausspr      S4
```

Looking a bit more closely at what we released, *we haven't released anyone's data directly*: the command `dput` shows us exactly what is released verbatim, and it's clear that nobody's data is directly included in this:

```
dput(mod1)
```

```
## new("gausspr", tol = numeric(0), scaling = list(scaled = c(TRUE,
## FALSE, FALSE, FALSE, FALSE, FALSE, FALSE, TRUE, FALSE, FALSE,
## FALSE, FALSE, TRUE), x.scale = list(`scaled:center` = c(age = 29.64,
## prev_convictions = 1.18, prev_rehab = 1.66), `scaled:scale` = c(age = 6.25564235106914,
## prev_convictions = 1.22374467124812, prev_rehab = 1.52006981579727
## )), y.scale = list(`scaled:center` = 0.5, `scaled:scale` = 0.505076272276105)),
##     sol = structure(numeric(0), .Dim = c(0L, 0L)), alphaindex = list(),
##     nvar = numeric(0), alpha = structure(c(0.494194475985336,
##     0.504104719583325, 0.494975370432552, 0.494793634826664,
##     -0.475868325923097, -0.495157346699524, 0.494486065455878,
##     -0.496066461223181, -0.475896248799934, -0.494971531503884,
##     0.49496865457255, -0.494963738012464, 0.496898997759016,
##     0.498109833034276, 0.495126721198407, -0.49334379925321,
##     0.493741731688079, -0.495290313082235, -0.496274888678442,
##     0.495214858416443, -0.494973867072998, 0.494986490980358,
##     -0.495298492091012, -0.506388103152589, -0.498104390660131,
##     -0.496443904999153, 0.494527975607549, -0.495035556900245,
##     -0.494819934435864, 0.49504832631987, -0.495054254978252,
##     -0.465828581073045, -0.465873633772586, 0.494825177970834,
##     0.494500497558978, 0.494523702434531, -0.494505012533582,
##     0.496100821437075, -0.494915907373877, -0.495261742190387,
##     0.495278188034688, 0.497718678979343, 0.49497479127283, -0.494860693951191,
##     0.494987436860571, 0.49735083822366, 0.495084555000569, -0.492507960456503,
##     0.495140677238289, -0.494974774878679), .Dim = c(50L, 1L), .Dimnames = list(
##         c("1", "2", "3", "4", "5", "6", "7", "8", "9", "10",
##         "11", "12", "13", "14", "15", "16", "17", "18", "19",
##         "20", "21", "22", "23", "24", "25", "26", "27", "28",
```

```
##         "29", "30", "31", "32", "33", "34", "35", "36", "37",
##         "38", "39", "40", "41", "42", "43", "44", "45", "46",
##         "47", "48", "49", "50"), NULL)), type = "regression",
##     kernelf = new("rbfkernel", .Data = function (x, y = NULL)
##     {
##         if (!is(x, "vector"))
##             stop("x must be a vector")
##         if (!is(y, "vector") && !is.null(y))
##             stop("y must a vector")
##         if (is(x, "vector") && is.null(y)) {
##             return(1)
##         }
##         if (is(x, "vector") && is(y, "vector")) {
##             if (!length(x) == length(y))
##                 stop("number of dimension must be the same on both data points")
##             return(exp(sigma * (2 * crossprod(x, y) - crossprod(x) -
##                 crossprod(y))))
##         }
##     }, kpar = list(sigma = 3)), kpar = list(), xmatrix = structure(c(0.696970791377618,
##     0.856826477473402, 1.65610490795232, 1.01668216356919, 0.37725941918605,
##     -1.22129744177179, -0.42201901129287, -1.70086450005914,
##     -0.262163325197086, 0.856826477473402, 0.0575480469944821,
##     -1.06144175567601, 0.537115105281834, 1.65610490795232, -1.38115312786757,
##     -0.741730383484438, -0.262163325197086, -0.102307639101302,
##     -0.42201901129287, -0.581874697388654, 0.696970791377618,
##     -0.581874697388654, -0.901586069580222, 1.17653784966497,
##     1.81596059404811, -0.581874697388654, 1.01668216356919, 1.81596059404811,
##     0.0575480469944821, 2.13567196623967, -0.581874697388654,
##     0.217403733090266, 0.37725941918605, -1.38115312786757, -0.581874697388654,
##     1.33639353576075, -0.42201901129287, 0.37725941918605, 0.0575480469944821,
##     1.01668216356919, -0.741730383484438, 0.537115105281834,
##     -1.22129744177179, -2.02057587225071, -0.581874697388654,
##     -1.38115312786757, 0.0575480469944821, -1.06144175567601,
##     -0.102307639101302, -0.102307639101302, 0, 0, 0, 0, 1, 0,
##     0, 1, 1, 1, 0, 0, 1, 1, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 1,
##     1, 0, 1, 0, 1, 0, 0, 0, 1, 0, 0, 1, 0, 1, 0, 0, 0, 0, 1,
##     0, 1, 1, 1, 0, 0, 1, 0, 0, 0, 0, 1, 1, 0, 0, 0, 1, 0, 0,
##     0, 1, 0, 1, 0, 0, 0, 1, 1, 0, 0, 0, 0, 1, 0, 0, 0, 1, 0,
##     0, 0, 1, 1, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 1, 0, 0,
##     1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 1,
##     0, 0, 1, 1, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
##     0, 1, 1, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 0, 0, 0, 0,
##     0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0,
##     0, 0, 0, 0, 1, 1, 0, 0, 0, 0, 1, 0, 1, 0, 0, 0, 0, 0, 0, 0,
##     0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
##     0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
##     0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 1,
##     0, 0, 0, 1, 1, 0, 0, 1, 0, 1, 1, 0, 1, 0, 1, 0, 0, 1, 0,
##     0, 1, 1, 0, 1, 1, 1, 1, 1, 0, 0, 0, 0, 1, 1, 0, 0, 1, 1,
##     1, 0, 1, 1, 0, 1, 1, 1, 0, 0.670074419334279, -0.964253432700547,
##     0.670074419334279, 1.48723834535169, -0.147089506683134,
##     -0.964253432700547, -0.964253432700547, -0.147089506683134,
##     -0.147089506683134, -0.147089506683134, 0.670074419334279,
```

```
##     1.48723834535169, -0.964253432700547, -0.964253432700547,
##     -0.964253432700547, -0.964253432700547, 0.670074419334279,
##     -0.964253432700547, 0.670074419334279, -0.964253432700547,
##     -0.964253432700547, 1.48723834535169, -0.147089506683134,
##     -0.964253432700547, -0.964253432700547, 1.48723834535169,
##     2.3044022713691, -0.147089506683134, -0.964253432700547,
##     -0.964253432700547, 1.48723834535169, -0.147089506683134,
##     -0.964253432700547, 0.670074419334279, 0.670074419334279,
##     2.3044022713691, -0.964253432700547, 1.48723834535169, 0.670074419334279,
##     -0.147089506683134, -0.964253432700547, -0.147089506683134,
##     -0.964253432700547, 0.670074419334279, 0.670074419334279,
##     0.670074419334279, 0.670074419334279, -0.964253432700547,
##     -0.964253432700547, -0.964253432700547, 0, 0, 0, 0, 0, 0,
##     1, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0,
##     0, 0, 0, 0, 0, 0, 1, 1, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0,
##     0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 1, 0, 1, 0, 0, 0, 0, 0,
##     0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 1, 0, 1, 0, 0, 0, 1, 0, 0,
##     0, 0, 1, 1, 0, 0, 0, 1, 0, 1, 0, 1, 1, 1, 1, 0, 1, 1, 1,
##     0, 1, 1, 0, 0, 0, 0, 0, 1, 0, 1, 1, 0, 0, 1, 1, 0, 0, 0,
##     1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 1, 0, 0,
##     0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0,
##     1, 0, 0, 0, 0, 1, 1, 0, 0, 0, 1, 0, 0, 1, 0, 0, 0, 0, 1,
##     1, 1, 0, 1, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 1, 0, 0, 0,
##     0, 0, 0, 0, -1.09205510348834, -0.434190583314643, 1.53940297720646,
##     -0.434190583314643, 0.223673936859058, -1.09205510348834,
##     0.223673936859058, -0.434190583314643, -0.434190583314643,
##     1.53940297720646, 0.223673936859058, -1.09205510348834, 0.223673936859058,
##     0.881538457032759, -0.434190583314643, 0.223673936859058,
##     -1.09205510348834, 0.223673936859058, -1.09205510348834,
##     -0.434190583314643, 1.53940297720646, -0.434190583314643,
##     -1.09205510348834, -0.434190583314643, 1.53940297720646,
##     0.223673936859058, 1.53940297720646, -1.09205510348834, 0.881538457032759,
##     -1.09205510348834, 0.881538457032759, -1.09205510348834,
##     -1.09205510348834, 0.223673936859058, -1.09205510348834,
##     1.53940297720646, -1.09205510348834, -0.434190583314643,
##     0.223673936859058, -1.09205510348834, -1.09205510348834,
##     -1.09205510348834, 2.19726749738016, 0.881538457032759, 0.881538457032759,
##     -0.434190583314643, -0.434190583314643, 0.881538457032759,
##     0.223673936859058, 2.19726749738016), .Dim = c(50L, 13L), .Dimnames = list(
##         c("1", "2", "3", "4", "5", "6", "7", "8", "9", "10",
##         "11", "12", "13", "14", "15", "16", "17", "18", "19",
##         "20", "21", "22", "23", "24", "25", "26", "27", "28",
##         "29", "30", "31", "32", "33", "34", "35", "36", "37",
##         "38", "39", "40", "41", "42", "43", "44", "45", "46",
##         "47", "48", "49", "50"), c("age", "educationNone", "educationSome_secondary",
##         "educationSecondary", "educationUniversity", "educationProfessional",
##         "sexF", "prev_convictions", "housingRent", "housingSupported",
##         "housingOther", "housingNo_fixed", "prev_rehab"))), ymatrix = NULL,
##     fitted = structure(c(0.49575501767583, 0.485844774077841,
##     0.494974123228614, 0.495155858834503, -0.51408116773807,
##     -0.494792146961643, 0.495463428205288, -0.493883032437986,
##     -0.514053244861233, -0.494977962157282, 0.494980839088616,
##     -0.494985755648702, 0.493050495902151, 0.49183966062689,
```

```
##     0.494822772462759, -0.496605694407956, 0.496207761973087,
##     -0.494659180578931, -0.493674604982725, 0.494734635244724,
##     -0.494975626588168, 0.494963002680809, -0.494651001570154,
##     -0.483561390508578, -0.491845103001036, -0.493505588662013,
##     0.495421518053617, -0.494913936760921, -0.495129559225303,
##     0.494901167341296, -0.494895238682915, -0.524120912588122,
##     -0.52407585988858, 0.495124315690332, 0.495448996102188,
##     0.495425791226635, -0.495444481127585, 0.493848672224091,
##     -0.49503358628729, -0.494687751470779, 0.494671305626479,
##     0.492230814681823, 0.494974702388337, -0.495088799709976,
##     0.494962056800596, 0.492598655437506, 0.494864938660597,
##     -0.497441533204663, 0.494808816422877, -0.494974718782487
##     ), .Dim = c(50L, 1L)), lev = logical(0), nclass = 0L, error = 0.24377893993342,
##     cross = -1, n.action = NULL, terms = solvent ~ age + education +
##         sex + prev_convictions + housing + prev_rehab, kcall = .local(x = x,
##         data = ..1, kpar = ..2))
```

Now someone outside of the TRE can use the model to make predictions for new people, and hopefully help them avoid insolvency.

Attacker - what we know

Suppose now that we are an adversary... (job interview)

Some people have made public that they were part of the study: their information is as follows:

```
some_people_in_study=data_matrix[sample(3:30,20),]
some_people_NOT_in_study=new_data[sample(3:30,20),]
```

```
head(some_people_in_study)
```

```
##        names age      education sex prev_convictions   housing prev_rehab solvent
## 27      John  36 Some_secondary   F                4  No_fixed          4       1
## 20    Ashley  26      Secondary   M                0       Own          1       1
## 8  Elizabeth  19           None   F                1 Supported          1       0
## 11    Angela  30 Some_secondary   F                2      Rent          2       1
## 13     Tyler  33           None   F                0     Other          2       1
## 21     Jason  34 Some_secondary   F                0     Other          4       0
```

We also know the details of some people who were not in the study:

```
head(some_people_NOT_in_study)
```

```
##        names age      education sex prev_convictions   housing prev_rehab solvent
## 22 Christina  35           None   F                4 Supported          3       0
## 7       Jose  31 Some_secondary   F                4       Own          0       1
## 16    Thomas  22      Secondary   M                0     Other          2       1
## 18  Brittany  24           None   M                1 Supported          3       0
## 19  Nicholas  27           None   M                1     Other          0       1
## 9      Megan  29           None   F                1 Supported          2       0
```

Attacker - what we can do

We have four potential candidates from a job interview,

```
candidates
```

```
##      names age      education sex prev_convictions   housing prev_rehab solvent
## 1 Michelle  34 Some_secondary   F                2     Other          0       1
## 2  Lindsay  35      Secondary   M                0 Supported          1       1
## 3  Zachary  24     University   M                5      Rent          0       1
## 4     Sean  20     University   M                4  No_fixed          0       1
```

The job interview candidates will not tell us if they are IV drug users. Can we use the released model to find out if they were in the study

Attack procedure

Let's say we use the model released from the TRE to make predictions on the people we *know* were in the dataset.

```
p1=predict(mod1,newdata=some_people_in_study,type="response")
head(data.frame(names=some_people_in_study$names,model_output=p1))
```

```
##       names model_output
## 1      John    0.7502257
## 2    Ashley    0.7498787
## 3 Elizabeth    0.2505514
## 4    Angela    0.7500031
## 5     Tyler    0.7490281
## 6     Jason    0.2499996
```

and the people we know were *not* in the study

```
p2=predict(mod1,newdata=some_people_NOT_in_study,type="response")
head(data.frame(names=some_people_NOT_in_study$names,model_output=p2))
```

```
##       names model_output
## 1 Christina    0.5000843
## 2      Jose    0.5000056
## 3    Thomas    0.5010153
## 4  Brittany    0.4999616
## 5  Nicholas    0.4982589
## 6     Megan    0.5084548
```

A pattern is suddenly obvious! People in the study have predictions which are very close to 25% or 75% People who were not have predictions close to 50%.

So now we make predictions on our job interview candidates:

```
p3=predict(mod1,newdata=candidates,type="response")
head(data.frame(names=candidates$names,model_output=p3))
```

```
##      names model_output
## 1 Michelle    0.7503941
## 2  Lindsay    0.7453887
## 3  Zachary    0.4999998
## 4     Sean    0.4999577
```

It's immediately obvious who was in the study and who was not:

```
candidate_status
```

```
##      names age              status
## 1 Michelle  34 In study; drug user
## 2  Lindsay  35 In study; drug user
```

```
## 3  Zachary  24        Not in study
## 4     Sean  20        Not in study
```

This is a violation of the privacy of the individuals (Michelle and Lindsay) who volunteered to be in the study.

Summary

Privacy was violated even though nothing malicious was done by researchers and no data was directly released.