# NHS Prescription Information from January and February 2016

Quinn Stevens

June 24, 2017

## Executive Summary

This report details the steps taken to analyse a dataset of NHS prescriptions in January and February 2016.

# Contents

# Chapter 1

# Introduction

In this report I will be downloading a dataset from the NHS website which keeps track of information about UK GP practices, their numbers of patients, the chemicals they prescribed, and the prescriptions they gave out in the period of January and February 2016.

I will import this data into a MySQL database and use SQL queries to answer the following questions:

1. How many practices and patients are registered in the N17 postcode?
2. Which practice prescribed the most beta blockers per registered patient in total over the two months?
3. Which was the most prescribed medication across all practices?
4. Which practice spent the most and least per patient?
5. What was the difference in selective serotonin reuptake inhibitor prescriptions between January and February?
6. Visualise the top 10 practices by number of metformin prescriptions throughout the entire period.

# Chapter 2

# Setting up the Database

In order to set up the database I downloaded the XAMPP database utility [1] so I could use its shell program to run MySQL on my computer. I set up XAMPP and my own user account (in order for some of the commands to work properly I needed to use the `mysqladmin.exe -u root password [password]` command to set a root password and then `mysql -u root -p` to log in using that password.

Once logged in to MySQL on the XAMPP shell I created the database using the `CREATE DATABASE prescriptionsdb;` command, then switched to that database with `\u prescriptionsdb`.

# Chapter 3

# Importing the Data

I downloaded the csv files to import from the NHS website [2, 3] and looked through them manually to wrk out how to set up the tables.

## 3.1 Setting up the tables

I decided to combine the data from January and February together, which left me with four tables to set up: `address` to keep track of practice names and addresses, `prescriptions` for data on the actual prescriptions, `chems` for details of the chemicals in use, and `patients` for the number of patients registered at each practice.

### 3.1.1 Chems table

For `chems`, the csv files were not set up in a way which allowed easy import, as they had three columns, but the third column wasn't named properly and wasn't populated. The three columns I made in my version of the table were `chem_sub`, `name` and `period`, to keep track of the chemical's unique ID, name, and the month the entry was from respectively. I set this up with the following command:

```
CREATE TABLE chems(Chem_Sub VARCHAR(255), Name
 ↪  VARCHAR(255), Period INT);
```

Listing 3.1: Creating the chems table

### 3.1.2 Patients table

For the `patients` table, the creation command was much longer, as there were many columns that weren't actually needed to answer the question. The command I used was:

```
CREATE TABLE patients(Practice_ID VARCHAR(255), Postcode
 ↪ VARCHAR(255), ONS_CCG_Code VARCHAR(255), CCG_Code
 ↪ VARCHAR(255), ONS_Region_Code VARCHAR(255),
 ↪ NHSE_Region_Code VARCHAR(255), ONS_Comm_Rgn_Code
 ↪ VARCHAR(255), NHSE_Comm_Rgn_Code VARCHAR(255),
 ↪ Total_All INT, Total_Male INT, Total_Female INT,
 ↪ Male_0_4 INT, Male_5_9 INT, Male_10_14 INT,
 ↪ Male_15_19 INT, Male_20_24 INT, Male_25_29 INT,
 ↪ Male_30_34 INT, Male_35_39 INT, Male_40_44 INT,
 ↪ Male_45_49 INT, Male_50_54 INT, Male_55_59 INT,
 ↪ Male_60_64 INT, Male_65_69 INT, Male_70_74 INT,
 ↪ Male_75_79 INT, Male_80_84 INT, Male_85_89 INT,
 ↪ Male_90_94 INT, Male_95_Up INT, Female_0_4 INT,
 ↪ Female_5_9 INT, Female_10_14 INT, Female_15_19 INT,
 ↪ Female_20_24 INT, Female_25_29 INT, Female_30_34 INT,
 ↪ Female_35_39 INT, Female_40_44 INT, Female_45_49 INT,
 ↪ Female_50_54 INT, Female_55_59 INT, Female_60_64 INT,
 ↪ Female_65_69 INT, Female_70_74 INT, Female_75_79 INT,
 ↪ Female_80_84 INT, Female_85_89 INT, Female_90_94 INT,
 ↪ Female_95_Up INT);
```

Listing 3.2: Creating the patients table

This created a very bloated table, which I later trimmed down to the essentials once the data had been imported.

### 3.1.3 Prescriptions table

The `prescriptions` table was relatively simple to create, as all the necessary columns were marked clearly in the csv file. I used this command to create the table:

```
CREATE TABLE prescriptions(SHA VARCHAR(255), PCT
 ↪ VARCHAR(255), Practice_ID VARCHAR(255), BNF_Code
 ↪ VARCHAR(255), BNF_Name VARCHAR(255), Items INT, NIC
 ↪ DOUBLE, ACT_Cost DOUBLE, Quantity INT, Period INT);
```

Listing 3.3: Creating the prescriptions table

### 3.1.4 Address table

Finally, for the `address` table I needed to use my own initiative to come up with the column names, as they weren't labelled in the csv file. Most of them were obvious, but specifically I chose to label the columns containing the practice

address as `Address_1` through `Address_4` as it wasn't consistent which part of the address was stored in each column, so naming them `Street`, `Town` etc would not have worked. In the end I used this command:

```
CREATE TABLE address(Period VARCHAR(255), Practice_ID
 ↪ VARCHAR(255), Practice_Name VARCHAR(255), Address_1
 ↪ VARCHAR(255), Address_2 VARCHAR(255), Address_3
 ↪ VARCHAR(255), Address_4 VARCHAR(255), Postcode
 ↪ VARCHAR(255));
```

Listing 3.4: Creating the address table

## 3.2 Populating and cleaning up the tables

Populating the `prescriptions`, `address` and `patients` tables was easy; I simply used the following command for each (without using `IGNORE 1 LINES;` for `address`, as that csv did not have column labels):

```
LOAD DATA INFILE 'filename.csv'
INTO TABLE tablename
FIELDS TERMINATED BY ','
LINES TERMINATED BY '\n'
IGNORE 1 LINES;
```

Listing 3.5: Loading data into tables

Due to how bloated the `patients` table was with unnecessary data, I then used the command in listing 3.6 to trim it down; this made the table a lot easier to manage.

Finally, the `chems` table needed a little more manual manipulation. I imported the January file with `LOAD DATA INFILE '201601CHEM SUBS.csv' FIELDS TERMINATED BY ',' LINES TERMINATED BY '\n' IGNORE 1 LINES;`, then used `UPDATE chems SET Period=201601 WHERE Period=0;` to change the period to be correct. I then did the same with the february file, replacing 201601 with 201602.

```
ALTER TABLE patients DROP COLUMN ONS_CCG_Code, DROP
 ↪  COLUMN CCG_Code, DROP COLUMN ONS_Region_Code, DROP
 ↪  COLUMN NHSE_Region_Code, DROP COLUMN
 ↪  ONS_Comm_Rgn_Code, DROP COLUMN NHSE_Comm_Rgn_Code,
 ↪  DROP COLUMN Total_Male, DROP COLUMN Total_Female,
 ↪  DROP COLUMN Male_0_4, DROP COLUMN Male_5_9, DROP
 ↪  COLUMN Male_10_14, DROP COLUMN Male_15_19, DROP
 ↪  COLUMN Male_20_24, DROP COLUMN Male_25_29, DROP
 ↪  COLUMN Male_30_34, DROP COLUMN Male_35_39, DROP
 ↪  COLUMN Male_40_44, DROP COLUMN Male_45_49, DROP
 ↪  COLUMN Male_50_54, DROP COLUMN Male_55_59, DROP
 ↪  COLUMN Male_60_64, DROP COLUMN Male_65_69, DROP
 ↪  COLUMN Male_70_74, DROP COLUMN Male_75_79, DROP
 ↪  COLUMN Male_80_84, DROP COLUMN Male_85_89, DROP
 ↪  COLUMN Male_90_94, DROP COLUMN Male_95_Up, DROP
 ↪  COLUMN Female_0_4, DROP COLUMN Female_5_9, DROP
 ↪  COLUMN Female_10_14, DROP COLUMN Female_15_19, DROP
 ↪  COLUMN Female_20_24, DROP COLUMN Female_25_29, DROP
 ↪  COLUMN Female_30_34, DROP COLUMN Female_35_39, DROP
 ↪  COLUMN Female_40_44, DROP COLUMN Female_45_49, DROP
 ↪  COLUMN Female_50_54, DROP COLUMN Female_55_59, DROP
 ↪  COLUMN Female_60_64, DROP COLUMN Female_65_69, DROP
 ↪  COLUMN Female_70_74, DROP COLUMN Female_75_79, DROP
 ↪  COLUMN Female_80_84, DROP COLUMN Female_85_89, DROP
 ↪  COLUMN Female_90_94, DROP COLUMN Female_95_Up;
```

Listing 3.6: Trimming down the patients table

# Chapter 4

# Making the Queries

## 4.1 Question 1: How many practices and registered patients are there in the N17 postcode area?

**Interpretation** I interpreted this question to be asking for the number of practices with a postcode beginning with 'N17', and the total number of patients registered at those practices.

```
SELECT COUNT(Postcode) AS practices_in_area,
 ↪  SUM(Total_all) AS patients_in_area FROM patients
 ↪  WHERE POSTCODE LIKE 'N17%';
```

Listing 4.1: Question 1 query

**Results**

| practices_in_area | patients_in_area |
|---|---|
| 7 | 52248 |

**Conclusion** There are 7 practices in the N17 postal area, with a total of 52,248 patients registered between them.

## 4.2 Question 2: Which practice prescribed the most beta-blockers per registered patient in total over the two month period?

**Interpretation** I interpreted this question to be asking for which practice prescribed the largest *quantity* of beta-blockers per patient registered at that practice. The beta blockers I searched for were listed on the NHS website [4]:

- atenolol
- bisoprolol
- carvedilol
- metroprolol
- nebivolol
- propranolol
- tenormin
- cardicor
- emcor
- betaloc
- lopresor
- nebilet
- inderal

```
SELECT
    (SELECT address.practice_name FROM address WHERE
     ↪ address.Practice_ID = prescriptions.PRACTICE_ID
     ↪ group by practice_id) AS Practice_name,
    prescriptions.practice_id AS practice_id,
        (SELECT Total_All FROM patients WHERE
         ↪ patients.Practice_ID =
         ↪ prescriptions.Practice_ID) AS 'patients',
    (SUM(quantity) / (SELECT Total_All FROM patients
     ↪ WHERE patients.Practice_id =
     ↪ prescriptions.practice_id)) AS
     ↪ 'betablockers/patient'
    FROM prescriptions
    WHERE bnf_name LIKE '%atenolol%'
        OR bnf_name LIKE '%bisoprolol%'
        OR bnf_name LIKE '%carvedilol%'
        OR bnf_name LIKE '%metroprolol%'
        OR bnf_name LIKE '%nebivolol%'
        OR bnf_name LIKE '%propranolol%'
        OR bnf_name LIKE '%tenormin%'
        OR bnf_name LIKE '%cardicor%'
        OR bnf_name LIKE '%emcor%'
        OR bnf_name LIKE '%betaloc%'
        OR bnf_name LIKE '%lopresor%'
        OR bnf_name LIKE '%nebilet%'
        OR bnf_name LIKE '%inderal%'
    GROUP BY Practice_ID
    ORDER BY betablockers_per_patient DESC
    LIMIT 1;
```

Listing 4.2: Question 2 query

**Results**

| Practice Name | Practice ID | Patients | Betablockers/patient |
|---|---|---|---|
| Burrswood Nursing Home | G82651 | 1 | 161.0000 |

**Conclusion** The practice which prescribed the most beta-blockers per patient was burrswood nursing home. This seems to be because there is only one registered patient at burrswood nursing home.

## 4.3 Question 3: Which was the most prescribed medication across all practices?

**Interpretation** I interpreted this question to be asking for the greatest number of separate prescriptions (items) given for an individual chemical (`chem_sub` in the `chems` table, or the first nine digits of `bnf_code` in the `prescriptions` table)

```
SELECT
    (SELECT name
        FROM chems
        WHERE SUBSTR(prescriptions.bnf_code, 1,
         ↪  9)=chems.chem_sub
        GROUP BY SUBSTR(prescriptions.bnf_code, 1, 9))
    AS chem_name,
    COUNT(items) AS items_prescribed
    FROM prescriptions
    GROUP BY SUBSTR(prescriptions.bnf_code, 1, 9)
    ORDER BY items_prescribed DESC
    LIMIT 1;
```

Listing 4.3: Question 3 query

**Results**

| Chem Name | Items Prescribed |
|---|---|
| Colecalciferol | 280495 |

**Conclusion** The most prescribed drug across all practices in January & February 2016 was Colecalciferol.

## 4.4 Question 4: Which practice spent the most and least per patient?

**Interpretation** I interpreted this question to be asking for the practices with the highest and lowest values of `SUM(act_cost) / total_all`, or: actual cost of all prescriptions divided by number of patients registered at that practice.

**Queries** This query returns many results with `total_all = NULL`, and I couldn't find a way to exlude these results from the query, so I used the first result and the last non-NULL result as my answers to this question.

```
SELECT
    (SELECT practice_name
        FROM address
        WHERE address.practice_id =
         ↪ prescriptions.practice_id
        GROUP BY practice_id)
    AS Name,
    SUM(act_cost) AS 'Total Cost',
    (SELECT Total_All
        FROM patients
        WHERE
         ↪ patients.practice_ID=prescriptions.practice_ID
        GROUP BY practice_ID)
    AS Patients,
    SUM(act_cost) / (SELECT Total_all
        FROM patients
        WHERE
         ↪ patients.practice_ID=prescriptions.practice_id
        GROUP BY practice_id)
    AS 'Price/patient'
    FROM prescriptions
    GROUP BY practice_ID
    ORDER BY price_per_patient DESC;
```

Listing 4.4: Question 4 Query

**Results**

| Name | Total Cost | Patients | Price/patient |
|---|---|---|---|
| Burrswood Nursing Home | 7609.05 | 1 | 7609.05 |
| ... | ... | ... | ... |
| School Lane PMS Practice | 50.74 | 3777 | 0.013 |
| Ashgate Hospice Daycentre | 1843.03 | NULL | NULL |
| ... | ... | ... | ... |

**Conclusion** The practice which spent most per patient was Burrswood Nursing Home, and the one which spent the least was School Lane PMS Practice. Burrswood turns up once again because it has only one patient.

## 4.5 Question 5: What was the difference in selective serotonin reuptake inhibitor prescriptions between January and February?

**Interpretation** I interpreted this question to be asking to work out the *quantity* of SSRIs prescribed in January and in February and to then find the difference. The list of SSRIs I searched for, from the NHS website [5]:

- citalopram
- cipramil
- dapoxetine
- priligy
- excitalopram
- cipralex
- fluoxetine
- prozac
- oxactin
- fluvoxamine
- faverin
- paroxetine
- seroxat
- sertraline
- lustral

I decided to use views for this in the hope that it would speed up the search process - it worked quite well, but this was the only question I could get views to work on.

```
CREATE VIEW ssris_jan AS
        SELECT bnf_name NAME,
        items ITEMS,
        period PERIOD
        FROM prescriptions
        WHERE period=201601
                AND (bnf_name LIKE '%citalopram%'
                OR bnf_name LIKE '%cipramil%'
                OR bnf_name LIKE '%dapoxetine%'
                OR bnf_name LIKE '%priligy%'
                OR bnf_name LIKE '%escitalopram%'
                OR bnf_name LIKE '%cipralex%'
                OR bnf_name LIKE '%fluoxetine%'
                OR bnf_name LIKE '%prozac%'
                OR bnf_name LIKE '%oxactin%'
                OR bnf_name LIKE '%fluvoxamine%'
                OR bnf_name LIKE '%faverin%'
                OR bnf_name LIKE '%paroxetin%'
                OR bnf_name LIKE '%seroxat%'
                OR bnf_name LIKE '%sertraline%'
                OR bnf_name LIKE '%lustral%');
```

Listing 4.5: Question 5 - Creating View for January SSRIs

```
CREATE VIEW ssris_feb AS
        SELECT bnf_name NAME,
        items ITEMS,
        period PERIOD
        FROM prescriptions
        WHERE period=201602
                AND (bnf_name LIKE '%citalopram%'
                OR bnf_name LIKE '%cipramil%'
                OR bnf_name LIKE '%dapoxetine%'
                OR bnf_name LIKE '%priligy%'
                OR bnf_name LIKE '%escitalopram%'
                OR bnf_name LIKE '%cipralex%'
                OR bnf_name LIKE '%fluoxetine%'
                OR bnf_name LIKE '%prozac%'
                OR bnf_name LIKE '%oxactin%'
                OR bnf_name LIKE '%fluvoxamine%'
                OR bnf_name LIKE '%faverin%'
                OR bnf_name LIKE '%paroxetin%'
                OR bnf_name LIKE '%seroxat%'
                OR bnf_name LIKE '%sertraline%'
                OR bnf_name LIKE '%lustral%');
```

Listing 4.6: Question 5 - Creating View for February SSRIs

```
SELECT SUM(items) AS January_SSRIs FROM ssris_jan;
```

Listing 4.7: Question 5 - Finding SSRIs Prescribed in January

```
SELECT SUM(items) AS February_SSRIs FROM ssris_feb;
```

Listing 4.8: Question 5 - Finding SSRIs Prescribed in February

**Results**

| January SSRIs |
|---|
| 2742049 |

| February SSRIs |
|---|
| 2725157 |

2742049 - 2725157 = 16892

**Conclusion** 16,892 fewer SSRIs were prescribed in February 2016 than in January 2016.

## 4.6 Question 6: Visualise the top 10 practices by number of metformin prescriptions throughout the entire period

**Interpretation** I interpreted this question to be asking for a table showing the name, id, postcode and metformin prescribed of the top 10 practices by metformin prescribed.

**Queries** At first I tried using views to answer this query, but I was not able to do so without the views being corrupted and taking a huge amount of time to query, so I decided to use nested SELECT statements instead.

```
SELECT
        (SELECT address.practice_name
                FROM address
                WHERE address.practice_id =
                 ↪ prescriptions.practice_id GROUP BY
                 ↪ practice_id)
        AS practice_name,
        SUM(items) AS metformin_prescribed
        FROM prescriptions
        WHERE
                bnf_name LIKE '%metformin%'
        GROUP BY prescriptions.practice_id
        ORDER BY metformin_prescribed DESC
        LIMIT 10;
```

Listing 4.9: Question 6 Query

**Results**

| Practice Name | Metformin Prescribed |
|---|---|
| Midlands Medical Partnership | 3192 |
| Lakeside Healthcare | 2848 |
| Spinney Hill Medical Centre | 2810 |
| The Shrewsbury Centre | 2739 |
| Beacon Medical Practice | 2652 |
| Marisco Medical Practice | 2545 |
| Vida Healthcare | 2264 |
| Harford Health Centre | 2183 |
| Portsdown Group Practice | 2108 |
| Coastal Medical Group | 2083 |

# Chapter 5

# Observations on the Dataset

Firstly, I think this dataset was quite messy, and required a significant amount of cleaning. The `address` file did not have any column titles, which meant some guesswork was required to decide what to name the columns in the table.

The `chem subs` file was also badly formatted. As far as I could tell, it was designed to have three columns, `CHEM_SUB`, `NAME` and `PERIOD`, to keep track of the chemical's ID, name, and the month in which the record was being kept. However, only `CHEM_SUB` and `NAME` were actually in the file. Where the `PERIOD` header should have been it instead said the month, which should have been in the fields rather than the column name. The fields themselves, meanwhile, were all empty.

If I was doing this project again, I would probably avoid combining multiple month's csv files into a single table, as it made it a lot harder by adding duplicates of what should have been the unique IDs of rows. On the other hand, combining multiple files into one table made it much easier to answer most of the questions without having to access multiple tables.

# Chapter 6

# Conclusions

In this report I used SQL queries to answer the following questions about an NHS database of GP practices, registered patients, chemicals and prescriptions from January and February 2016:

**Q:** How many practices and patients are registered in the N17 postcode?
**A:** There are 7 practices in the N17 postal area, with a total of 52,248 patients registered between them.

**Q:** Which practice prescribed the most beta blockers per registered patient in total over the two months?
**A:** The practice which prescribed the most beta-blockers per patient was burrswood nursing home. This seems to be because there is only one registered patient at burrswood nursing home.

**Q:** Which was the most prescribed medication across all practices?
**A:** The most prescribed drug across all practices in January & February 2016 was Colecalciferol.

**Q:** Which practice spent the most and least per patient?
**A:** The practice which spent most per patient was Burrswood Nursing Home, and the one which spent the least was School Lane PMS Practice. Burrswood turns up once again because it has only one patient.

**Q:** What was the difference in selective serotonin reuptake inhibitor prescriptions between January and February?
**A:** 16,892 fewer SSRIs were prescribed in February 2016 than in January 2016.

**Q:** Visualise the top 10 practices by number of metformin prescriptions throughout the entire period.

**A:**

| Practice Name | Metformin Prescribed |
|---|---|
| Midlands Medical Partnership | 3192 |
| Lakeside Healthcare | 2848 |
| Spinney Hill Medical Centre | 2810 |
| The Shrewsbury Centre | 2739 |
| Beacon Medical Practice | 2652 |
| Marisco Medical Practice | 2545 |
| Vida Healthcare | 2264 |
| Harford Health Centre | 2183 |
| Portsdown Group Practice | 2108 |
| Coastal Medical Group | 2083 |

# List of Listings

# Bibliography

[1] XAMPP (2017) Available at:
https://www.apachefriends.org/index.html

[2] NHS (2016) Available at:
https://data.gov.uk/dataset/prescribing-by-gp-practice-presentation-level

[3] NHS(2016) Available at:
https://data.gov.uk/dataset/numbers_of_patients_registered_at_a_gp_practice

[4] NHS (2017) *Beta-blockers*. Available at:
http://www.nhs.uk/conditions/beta-blockers/pages/introduction.aspx

[5] NHS (2017) *SSRIs*. Available at:
http://www.nhs.uk/conditions/SSRIs-(selective-serotonin-reuptake-inhibitors)/Pages/Introduction.aspx