What can EPC data tell us about the domestic cost of carbon?

Ben Anderson

Last run at: 2020-11-05 19:01:46

# Energy Performance Certificates (EPCs)

Apart from a few exempted buildings, a dwelling must have an EPC when constructed, sold or let. This means that over time we will have an EPC for an increasing number of properties and we should *already* have EPCs for all rented properties.

EPCs are not necessarily up to date. For example if a property has not been sold or let since a major upgrade, the effects of that upgrade may not be visible in the data.

Further reading:

* <https://epc.opendatacommunities.org/docs/guidance#technical_notes>
* <https://en.wikipedia.org/wiki/Energy_Performance_Certificate_(United_Kingdom)#Procedure>

check what feeds in automatically e.f. RHI installs etc

We have to assume the data we have is the *current state of play* for these dwellings.

# Southampton EPCs

df <- path.expand("~/data/EW\_epc/domestic-E06000045-Southampton/certificates.csv")  
allEPCs\_DT <- data.table::fread(df)

## Warning in require\_bit64\_if\_needed(ans): Some columns are type 'integer64' but package bit64 is not installed.  
## Those columns will print as strange looking floating point data. There is no need to reload the data. Simply  
## install.packages('bit64') to obtain the integer64 print method and print the data again.

The EPC data file has 91833 records for Southampton and 90 variables. We’re not interested in all of these, we want:

* PROPERTY\_TYPE: Describes the type of property such as House, Flat, Maisonette etc. This is the type differentiator for dwellings;
* BUILT\_FORM: The building type of the Property e.g. Detached, Semi-Detached, Terrace etc. Together with the Property Type, the Build Form produces a structured description of the property;
* ENVIRONMENT\_IMPACT\_CURRENT: A measure of the property’s current impact on the environment in terms of carbon dioxide (CO₂) emissions. The higher the rating the lower the CO₂ emissions. (CO₂ emissions in tonnes / year) **NB this is a categorised scale calculated from CO2\_EMISSIONS\_CURRENT**;
* ENERGY\_CONSUMPTION\_CURRENT: Current estimated total energy consumption for the property in a 12 month period (**kWh/m2**). Displayed on EPC as the current primary energy use per square metre of floor area. **Nb: this covers heat and hot water (and lightng?)**
* CO2\_EMISSIONS\_CURRENT: CO₂ emissions per year in tonnes/year **NB: this is calculated from the modelled kWh energy input using (possibly) outdated carbon intensity values**;
* TENURE: Describes the tenure type of the property. One of: Owner-occupied; Rented (social); Rented (private).

We’re also going to keep:

* WIND\_TURBINE\_COUNT: Number of wind turbines; 0 if none;
* PHOTO\_SUPPLY: Percentage of photovoltaic area as a percentage of total roof area. 0% indicates that a Photovoltaic Supply is not present in the property;
* TOTAL\_FLOOR\_AREA: The total useful floor area is the total of all enclosed spaces measured to the internal face of the external walls, i.e. the gross floor area as measured in accordance with the guidance issued from time to time by the Royal Institute of Chartered Surveyors or by a body replacing that institution. (m²) - to allow for the calculation of total energy demand;
* POSTCODE - to allow linkage to other datasets
* LOCAL\_AUTHORITY\_LABEL - for checking

These may indicate ‘non-grid’ energy inputs.

If an EPC has been updated or refreshed, the EPC dataset will hold multiple EPC records for that property. We will just select the most recent.

# select just these vars  
dt <- allEPCs\_DT[, .(BUILDING\_REFERENCE\_NUMBER, LMK\_KEY, LODGEMENT\_DATE, PROPERTY\_TYPE, BUILT\_FORM,  
 ENVIRONMENT\_IMPACT\_CURRENT, ENERGY\_CONSUMPTION\_CURRENT, CO2\_EMISSIONS\_CURRENT, TENURE,  
 PHOTO\_SUPPLY, WIND\_TURBINE\_COUNT, TOTAL\_FLOOR\_AREA,   
 POSTCODE, LOCAL\_AUTHORITY\_LABEL)]  
  
# select most recent record within BUILDING\_REFERENCE\_NUMBER - how?  
# better check this is doing so  
setkey(dt,BUILDING\_REFERENCE\_NUMBER, LODGEMENT\_DATE) # sort by date within reference number  
sotonUniqueEPCsDT <- unique(dt, by = "BUILDING\_REFERENCE\_NUMBER",  
 fromLast = TRUE) # which one does it take?  
  
test1 <- allEPCs\_DT[, .(min1 = min(LODGEMENT\_DATE),   
 nRecords = .N),   
 keyby = .(BUILDING\_REFERENCE\_NUMBER)]  
  
test2 <- sotonUniqueEPCsDT[, .(min2 = min(LODGEMENT\_DATE)),   
 keyby = .(BUILDING\_REFERENCE\_NUMBER)]  
t <- test1[test2]  
t[, diff := min2 - min1]  
  
summary(t[nRecords > 1]) # diff is always >= 0 so min2 (after unique) is always > min1

## BUILDING\_REFERENCE\_NUMBER min1 nRecords min2 diff   
## Min. : 0.000e+00 Min. :2008-10-01 Min. :2.000 Min. :2008-10-10 Min. : 0   
## 1st Qu.:1.224e-314 1st Qu.:2009-04-01 1st Qu.:2.000 1st Qu.:2014-08-22 1st Qu.: 745   
## Median :2.473e-314 Median :2010-03-01 Median :2.000 Median :2018-02-19 Median :1960   
## Mean :2.463e-314 Mean :2011-04-19 Mean :2.179 Mean :2017-01-05 Mean :2088   
## 3rd Qu.:3.703e-314 3rd Qu.:2013-03-18 3rd Qu.:2.000 3rd Qu.:2019-04-30 3rd Qu.:3643   
## Max. :4.940e-314 Max. :2020-06-30 Max. :9.000 Max. :2020-06-30 Max. :4279

# confirms fromLast = TRUE has selected the most recent within BUILDING\_REFERENCE\_NUMBER  
  
skimr::skim(sotonUniqueEPCsDT)

## Warning: Couldn't find skimmers for class: integer64; No user-defined `sfl` provided. Falling back to `character`.

Data summary

Name

sotonUniqueEPCsDT

Number of rows

71600

Number of columns

14

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Column type frequency:

character

7

Date

1

numeric

6

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Group variables

None

**Variable type: character**

skim\_variable

n\_missing

complete\_rate

min

max

empty

n\_unique

whitespace

BUILDING\_REFERENCE\_NUMBER

0

1

17

21

0

71600

0

LMK\_KEY

0

1

29

34

0

71600

0

PROPERTY\_TYPE

0

1

4

10

0

5

0

BUILT\_FORM

0

1

8

20

0

7

0

TENURE

0

1

0

16

1988

6

0

POSTCODE

0

1

8

8

0

5107

0

LOCAL\_AUTHORITY\_LABEL

0

1

11

11

0

1

0

**Variable type: Date**

skim\_variable

n\_missing

complete\_rate

min

max

median

n\_unique

LODGEMENT\_DATE

0

1

2008-10-01

2020-06-30

2014-10-20

4132

**Variable type: numeric**

skim\_variable

n\_missing

complete\_rate

mean

sd

p0

p25

p50

p75

p100

hist

ENVIRONMENT\_IMPACT\_CURRENT

0

1.00

62.56

15.73

1.0

52.0

63.0

73

115.00

▁▂▇▅▁

ENERGY\_CONSUMPTION\_CURRENT

0

1.00

262.90

140.55

-184.0

173.0

233.0

327

1597.00

▃▇▁▁▁

CO2\_EMISSIONS\_CURRENT

0

1.00

3.16

1.94

-1.8

1.8

2.8

4

77.00

▇▁▁▁▁

PHOTO\_SUPPLY

38595

0.46

0.59

5.12

0.0

0.0

0.0

0

100.00

▇▁▁▁▁

WIND\_TURBINE\_COUNT

5556

0.92

0.00

0.03

-1.0

0.0

0.0

0

1.00

▁▁▇▁▁

TOTAL\_FLOOR\_AREA

0

1.00

72.97

34.93

0.0

49.0

69.0

87

1353.68

▇▁▁▁▁

As we can see that we have 71600 unique property reference numbers. We can also see some strangeness. In some cases we seem to have:

* negative energy consumption;
* negative emissions;
* 0 floor area

This is not surprising since the kWh/y and TCO2/y values are estimated using a model but before we go any further we’d better check if these are significant in number.

## Check ‘missing’ EPC rates

We will do this mostly at MSOA level as it allows us to link to other MSOA level datasets. Arguably it would be better to do this at LSOA level but…

First we’ll use the BEIS 2018 MSOA level annual electricity data to estimate the number of meters (not properties) - some addresses can have 2 meters (e.g. standard & economy 7). This is more useful than the number of gas meters since not all dwellings have mains gas but all have an electricity meter.

beisElecDT <- data.table::fread("~/data/beis/MSOA\_DOM\_ELEC\_csv/MSOA\_ELEC\_2018.csv")  
sotonElecDT <- beisElecDT[LAName %like% "Southampton", .(nElecMeters = METERS,   
 beisElecMWh = KWH/1000,   
 MSOACode, LAName)  
 ]  
  
  
beisGasDT <- data.table::fread("~/data/beis/MSOA\_DOM\_GAS\_csv/MSOA\_GAS\_2018.csv")  
sotonGasDT <- beisGasDT[LAName %like% "Southampton", .(nGasMeters = METERS,   
 beisGasMWh = KWH/1000,   
 MSOACode)]  
  
setkey(sotonElecDT, MSOACode)  
setkey(sotonGasDT, MSOACode)  
sotonEnergyDT <- sotonGasDT[sotonElecDT]  
sotonEnergyDT[, beisEnergyMWh := beisElecMWh + beisGasMWh]  
#head(sotonEnergyDT)

Next we’ll check for the number of households reported by the 2011 Census.

would be better to use dwellings but this gives us tenure

#censusDT <- data.table::fread(path.expand("~/data/"))  
# IMD ----  
deprivationDT <- data.table::fread(path.expand("~/data/census2011/2011\_MSOA\_deprivation.csv"))  
deprivationDT[, totalHouseholds := `Household Deprivation: All categories: Classification of household deprivation; measures: Value`]  
deprivationDT[, MSOACode := `geography code`]  
setkey(deprivationDT, MSOACode)  
setkey(sotonElecDT, MSOACode)  
# link LA name from Soton elec for now  
sotonDep\_DT <- deprivationDT[sotonElecDT[, .(MSOACode, LAName)]]  
sotonDep\_DT[, nHHs\_deprivation := `Household Deprivation: All categories: Classification of household deprivation; measures: Value`]  
  
#sotonDep\_DT[, .(nHouseholds = sum(totalHouseholds)), keyby = .(LAName)]  
  
# census tenure ----  
sotonTenureDT <- data.table::fread(path.expand("~/data/census2011/2011\_MSOA\_householdTenure\_Soton.csv"))  
  
sotonTenureDT[, census2011\_socialRent := `Tenure: Social rented; measures: Value`]  
sotonTenureDT[, census2011\_privateRent := `Tenure: Private rented; measures: Value`]  
sotonTenureDT[, census2011\_ownerOccupy := `Tenure: Owned; measures: Value`]  
sotonTenureDT[, census2011\_other := `Tenure: Living rent free; measures: Value`]  
sotonTenureDT[, MSOACode := `geography code`]  
  
sotonTenureDT[, hhCheck := census2011\_socialRent + census2011\_privateRent + census2011\_ownerOccupy + census2011\_other]  
sotonTenureDT[, nHHs\_tenure := `Tenure: All households; measures: Value`]  
  
# summary(sotonTenureDT[, .(hhCheck, nHHs\_tenure)])  
# might not quite match due to cell perturbation etc?  
  
# join em ----  
setkey(sotonDep\_DT, MSOACode)  
setkey(sotonTenureDT, MSOACode)  
  
sotonCensus2011\_DT <- sotonTenureDT[sotonDep\_DT]  
  
t <- sotonCensus2011\_DT[, .(sum\_Deprivation = sum(nHHs\_deprivation),  
 sum\_Tenure = sum(nHHs\_tenure)), keyby = .(LAName)]  
kableExtra::kable(t, caption = "Census derived household counts")

Census derived household counts

LAName

sum\_Deprivation

sum\_Tenure

Southampton

98254

98254

That’s lower (as expected) but doesn’t allow for dwellings that were empty on census night.

# Postcodes don't help - no count of addresses in the data (there used to be??)  
# but we can use it to check which Soton postcodes are missing from the EPC file  
soPostcodesDT <- data.table::fread(path.expand("~/data/UK\_postcodes/NSPL\_AUG\_2020\_UK/Data/multi\_csv/NSPL\_AUG\_2020\_UK\_SO.csv"))  
  
soPostcodesDT <- soPostcodesDT[is.na(doterm)] # keep current  
  
sotonPostcodesDT <- soPostcodesDT[laua == "E06000045"] # keep Southampton City  
  
sotonPostcodesReducedDT <- sotonPostcodesDT[, .(pcd, pcd2, pcds, laua, msoa11, lsoa11)]  
  
sotonPostcodesReducedDT[, c("pc\_chunk1","pc\_chunk2" ) := tstrsplit(pcds,   
 split = " "  
 )  
 ]  
sotonPostcodesReducedDT[, .(nEPCs = .N), keyby = .(pc\_chunk1)]

## pc\_chunk1 nEPCs  
## 1: SO14 849  
## 2: SO15 1176  
## 3: SO16 1328  
## 4: SO17 443  
## 5: SO18 859  
## 6: SO19 1164

We should not have single digit postcodes in the postcode data - i.e. S01 should not be there (since 1993). Southampton City is unusual in only having [double digit postcodes](https://en.wikipedia.org/wiki/SO_postcode_area).

# EPC  
# set up counters  
sotonUniqueEPCsDT[, epcIsSocialRent := ifelse(TENURE == "rental (social)", 1, 0)]  
sotonUniqueEPCsDT[, epcIsPrivateRent := ifelse(TENURE == "rental (private)", 1, 0)]  
sotonUniqueEPCsDT[, epcIsOwnerOcc := ifelse(TENURE == "owner-occupied", 1, 0)]  
sotonUniqueEPCsDT[, epcIsUnknownTenure := ifelse(TENURE == "NO DATA!" |  
 TENURE == "" , 1, 0)]  
# aggregate EPCs to postcodes  
sotonEpcPostcodes\_DT <- sotonUniqueEPCsDT[, .(nEPCs = .N,  
 sumEPC\_tCO2 = sum(CO2\_EMISSIONS\_CURRENT, na.rm = TRUE),  
 n\_epcIsSocialRent = sum(epcIsSocialRent, na.rm = TRUE),  
 n\_epcIsPrivateRent = sum(epcIsPrivateRent, na.rm = TRUE),  
 n\_epcIsOwnerOcc = sum(epcIsOwnerOcc, na.rm = TRUE),  
 n\_epcIsUnknownTenure = sum(epcIsUnknownTenure, na.rm = TRUE),  
 sumEpcMWh = sum(ENERGY\_CONSUMPTION\_CURRENT\* TOTAL\_FLOOR\_AREA)/1000), # crucial as ENERGY\_CONSUMPTION\_CURRENT = kWh/m2  
 keyby = .(POSTCODE, LOCAL\_AUTHORITY\_LABEL)]  
  
sotonEpcPostcodes\_DT[, c("pc\_chunk1","pc\_chunk2" ) := tstrsplit(POSTCODE,   
 split = " "  
 )  
 ]  
sotonEpcPostcodes\_DT[, .(nEPCs = .N), keyby = .(pc\_chunk1)]

## pc\_chunk1 nEPCs  
## 1: SO14 601  
## 2: SO15 960  
## 3: SO16 1245  
## 4: SO17 403  
## 5: SO18 776  
## 6: SO19 1122

# check original EPC data for Soton - which postcodes are covered?  
allEPCs\_DT[, c("pc\_chunk1","pc\_chunk2" ) := tstrsplit(POSTCODE,   
 split = " "  
 )  
 ]  
allEPCs\_DT[, .(nEPCs = .N), keyby = .(pc\_chunk1)]

## pc\_chunk1 nEPCs  
## 1: SO14 14213  
## 2: SO15 17855  
## 3: SO16 20270  
## 4: SO17 8446  
## 5: SO18 10661  
## 6: SO19 20388

It looks like we have EPCs for each postcode sector which is good.

# match the EPC postcode summaries to the postcode extract  
sotonPostcodesReducedDT[, POSTCODE\_s := stringr::str\_remove(pcds, " ")]  
setkey(sotonPostcodesReducedDT, POSTCODE\_s)  
sotonPostcodesReducedDT[, MSOACode := msoa11]  
message("Number of postcodes: ",uniqueN(sotonPostcodesReducedDT$POSTCODE\_s))

## Number of postcodes: 5819

sotonEpcPostcodes\_DT[, POSTCODE\_s := stringr::str\_remove(POSTCODE, " ")]  
setkey(sotonEpcPostcodes\_DT, POSTCODE\_s)  
message("Number of postcodes with EPCs: ",uniqueN(sotonEpcPostcodes\_DT$POSTCODE\_s))

## Number of postcodes with EPCs: 5107

dt <- sotonEpcPostcodes\_DT[sotonPostcodesReducedDT]  
  
# aggregate to MSOA - watch for NAs where no EPCs in a given postcode  
sotonEpcMSOA\_DT <- dt[, .(nEPCs = sum(nEPCs, na.rm = TRUE),   
 sumEPC\_tCO2 = sum(sumEPC\_tCO2, na.rm = TRUE),  
 n\_epcIsSocialRent = sum(n\_epcIsSocialRent, na.rm = TRUE),  
 n\_epcIsPrivateRent = sum(n\_epcIsPrivateRent, na.rm = TRUE),  
 n\_epcIsOwnerOcc = sum(n\_epcIsOwnerOcc, na.rm = TRUE),  
 n\_epcIsUnknownTenure = sum(n\_epcIsUnknownTenure, na.rm = TRUE),  
 sumEpcMWh = sum(sumEpcMWh, na.rm = TRUE)  
 ),  
 keyby = .(MSOACode) # change name on the fly for easier matching  
 ]   
  
#summary(sotonEpcMSOA\_DT)

So we have some postcodes with no EPCs.

Join the estimates together at MSOA level for comparison. There are 32 MSOAs in Southampton.

# 32 LSOAs in Soton  
# add deprivation  
setkey(sotonEnergyDT, MSOACode)  
setkey(sotonCensus2011\_DT, MSOACode)  
setkey(sotonEpcMSOA\_DT, MSOACode)  
  
sotonMSOA\_DT <- sotonCensus2011\_DT[sotonEnergyDT]  
#names(sotonMSOA\_DT)  
sotonMSOA\_DT <- sotonEpcMSOA\_DT[sotonMSOA\_DT]  
#names(sotonMSOA\_DT)  
  
# add MSOA names from the postcode LUT  
  
msoaNamesDT <- data.table::as.data.table(readxl::read\_xlsx(path.expand("~/data/UK\_postcodes/NSPL\_AUG\_2020\_UK/Documents/MSOA (2011) names and codes UK as at 12\_12.xlsx")))  
msoaNamesDT[, MSOACode := MSOA11CD]  
msoaNamesDT[, MSOAName := MSOA11NM]  
setkey(msoaNamesDT, MSOACode)  
  
sotonMSOA\_DT <- msoaNamesDT[sotonMSOA\_DT]  
  
#names(sotonMSOA\_DT)

t <- sotonMSOA\_DT[, .(nHouseholds\_2011 = sum(nHHs\_tenure),  
 nElecMeters\_2018 = sum(nElecMeters),  
 nEPCs\_2020 = sum(nEPCs),  
 sumEPCMWh = sum(sumEpcMWh),  
 sumBEISMWh = sum(beisEnergyMWh),  
 sumEPC\_tCO2 = sum(sumEPC\_tCO2)  
 )]  
  
kableExtra::kable(t, caption = "Comparison of different estimates of the number of dwellings and energy demand") %>%  
 kable\_styling()

Comparison of different estimates of the number of dwellings and energy demand

nHouseholds\_2011

nElecMeters\_2018

nEPCs\_2020

sumEPCMWh

sumBEISMWh

sumEPC\_tCO2

98254

108333

71276

1286499

1276983

225568

nHouseholds\_2011f <- sum(sotonMSOA\_DT$nHHs\_tenure)  
nElecMeters\_2018f <- sum(sotonMSOA\_DT$elecMeters)  
nEPCs\_2020f <- sum(sotonMSOA\_DT$nEPCs)  
  
makePC <- function(x,y,r){  
 # make a percent of x/y and round it to r decimal places  
 pc <- round(100\*(x/y),r)  
 return(pc)  
}

We can see that the number of EPCs we have is:

* 72.5% of Census 2011 households
  + % of the recorded 2018 electricity meters

We can also see that despite having ‘missing’ EPCs, the estimated total EPC-derived energy demand is marginally higher than the BEIS-derived weather corrected energy demand data. Given that the BEIS data accounts for all heating, cooking, hot water, lighting and appliance use we would expect the EPC data to be lower *even if no EPCs were missing…*

sotonMSOA\_DT[, dep0\_pc := 100\*(`Household Deprivation: Household is not deprived in any dimension; measures: Value`/nHHs\_deprivation)]  
sotonMSOA\_DT[, socRent\_pc := 100\*(census2011\_socialRent/nHHs\_tenure)]  
sotonMSOA\_DT[, privRent\_pc := 100\*(census2011\_privateRent/nHHs\_tenure)]  
sotonMSOA\_DT[, ownerOcc\_pc := 100\*(census2011\_ownerOccupy/nHHs\_tenure)]  
  
t <- sotonMSOA\_DT[, .(MSOAName, MSOACode, nHHs\_tenure,nElecMeters,nEPCs,  
 dep0\_pc, socRent\_pc, privRent\_pc, ownerOcc\_pc,sumEpcMWh, beisEnergyMWh )]  
  
t[, pc\_missingHH := makePC(nEPCs,nHHs\_tenure,1)]  
t[, pc\_missingMeters := makePC(nEPCs,nElecMeters,1)]  
t[, pc\_energyBEIS := makePC(sumEpcMWh,beisEnergyMWh,1)]  
  
kableExtra::kable(t[order(-pc\_missingHH)], digits = 2, caption = "EPC records as a % of n census households and n meters per MSOA") %>%  
 kable\_styling()

EPC records as a % of n census households and n meters per MSOA

MSOAName

MSOACode

nHHs\_tenure

nElecMeters

nEPCs

dep0\_pc

socRent\_pc

privRent\_pc

ownerOcc\_pc

sumEpcMWh

beisEnergyMWh

pc\_missingHH

pc\_missingMeters

pc\_energyBEIS

Southampton 029

E02003577

4908

6734

5902

37.92

27.61

43.09

24.67

77239.37

47461.33

120.3

87.6

162.7

Southampton 023

E02003571

3040

3530

2958

47.96

20.23

56.97

21.48

47763.66

33485.69

97.3

83.8

142.6

Southampton 022

E02003570

3635

4142

3426

25.56

29.82

45.69

22.59

58314.50

49449.97

94.3

82.7

117.9

Southampton 017

E02003565

2563

2840

2403

48.81

11.63

57.24

28.95

42564.50

35191.63

93.8

84.6

121.0

Southampton 031

E02003579

3357

4460

3112

44.92

11.56

23.80

63.09

46142.79

47913.06

92.7

69.8

96.3

Southampton 013

E02003561

3181

3489

2663

39.80

19.68

44.73

33.98

49213.04

38430.85

83.7

76.3

128.1

Southampton 009

E02003557

2753

3103

2137

50.78

7.37

38.98

52.49

47035.21

42842.23

77.6

68.9

109.8

Southampton 020

E02003568

3820

3900

2959

50.08

4.53

50.92

42.93

53455.26

47024.09

77.5

75.9

113.7

Southampton 010

E02003558

2924

3222

2223

33.96

32.32

25.82

39.81

38646.22

34421.99

76.0

69.0

112.3

Southampton 021

E02003569

3527

3999

2671

40.71

15.00

38.28

44.32

46809.22

41380.67

75.7

66.8

113.1

Southampton 015

E02003563

3483

3818

2551

37.81

21.79

24.17

51.79

46440.94

39920.85

73.2

66.8

116.3

Southampton 027

E02003575

2808

2987

2028

29.59

51.14

8.23

37.64

36556.34

28941.46

72.2

67.9

126.3

Southampton 007

E02003555

3140

3763

2261

34.59

30.96

11.15

56.15

35182.76

40416.83

72.0

60.1

87.0

Southampton 005

E02003553

2394

2464

1686

39.01

25.44

40.27

32.00

31931.92

33476.91

70.4

68.4

95.4

Southampton 014

E02003562

3636

3921

2513

45.68

9.13

29.24

59.19

47112.87

51289.66

69.1

64.1

91.9

Southampton 032

E02003580

2617

2825

1786

27.21

55.48

6.65

35.69

30591.72

24488.16

68.2

63.2

124.9

Southampton 025

E02003573

3236

3470

2106

29.57

43.54

6.12

47.84

38983.67

41714.91

65.1

60.7

93.5

Southampton 003

E02003551

2256

2446

1456

33.69

38.96

15.29

42.95

27406.13

26316.61

64.5

59.5

104.1

Southampton 012

E02003560

3040

3191

1952

26.97

53.52

8.75

36.12

33850.25

34252.94

64.2

61.2

98.8

Southampton 006

E02003554

2646

2873

1684

46.49

14.55

21.05

63.00

35551.16

39712.77

63.6

58.6

89.5

Southampton 004

E02003552

2646

2809

1653

28.12

47.47

9.26

40.97

30037.47

28051.01

62.5

58.8

107.1

Southampton 028

E02003576

3434

3614

2121

38.99

22.83

18.58

56.41

39567.93

44100.48

61.8

58.7

89.7

Southampton 018

E02003566

2607

2831

1604

35.21

29.84

8.36

59.42

27084.70

36047.76

61.5

56.7

75.1

Southampton 016

E02003564

3474

3563

2124

39.38

22.54

12.09

63.39

42670.04

43718.49

61.1

59.6

97.6

Southampton 001

E02003549

2849

2832

1737

52.37

11.23

25.06

62.06

41456.28

49676.93

61.0

61.3

83.5

Southampton 002

E02003550

3216

3527

1923

43.10

21.05

11.04

66.08

36473.86

41124.17

59.8

54.5

88.7

Southampton 019

E02003567

2991

3200

1780

39.18

27.28

14.11

56.80

33242.61

43448.21

59.5

55.6

76.5

Southampton 026

E02003574

3412

3599

1972

40.77

11.78

13.66

71.72

37590.51

45562.64

57.8

54.8

82.5

Southampton 030

E02003578

2641

2830

1519

44.07

10.64

15.68

72.13

27635.37

36572.30

57.5

53.7

75.6

Southampton 024

E02003572

2484

2597

1367

45.61

8.13

15.46

75.28

30084.88

39680.38

55.0

52.6

75.8

Southampton 011

E02003559

3065

3165

1678

53.38

5.97

15.14

76.70

42375.46

55256.20

54.7

53.0

76.7

Southampton 008

E02003556

2471

2589

1321

42.57

12.51

16.15

70.42

27488.33

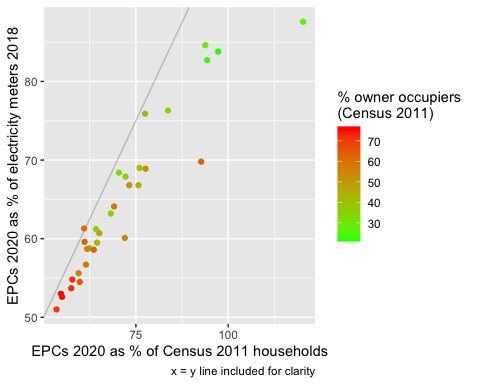
35612.03

53.5

51.0

77.2

ggplot2::ggplot(t, aes(x = pc\_missingHH,   
 y = pc\_missingMeters,  
 colour = round(ownerOcc\_pc))) +  
 geom\_abline(alpha = 0.2, slope=1, intercept=0) +  
 geom\_point() +  
 scale\_color\_continuous(name = "% owner occupiers \n(Census 2011)", high = "red", low = "green") +  
 #theme(legend.position = "bottom") +  
 labs(x = "EPCs 2020 as % of Census 2011 households",  
 y = "EPCs 2020 as % of electricity meters 2018",  
 caption = "x = y line included for clarity")



% ‘missing’ rates comparison

outlierMSOA <- t[pc\_missingHH > 100]

Figure @ref(tab:missingEPCbyMSOA) suggests that rates vary considerably by MSOA but are relatively consistent across the two baseline ‘truth’ estimates with the exception of E02003577 which appears to have many more EPCs than Census 2011 households. It is worth noting that [this MSOA](https://www.localhealth.org.uk/#c=report&chapter=c01&report=r01&selgeo1=msoa_2011.E02003577&selgeo2=eng.E92000001) covers the city centre and dock areas which have had substantial new build since 2011 and so may have households inhabiting dwellings that did not exist at Census 2011. This is also supported by the considerably higher EPC derived energy demand data compared to BEIS’s 2018 data - although it suggests the dwellings are either very new (since 2018) or are yet to be occupied.

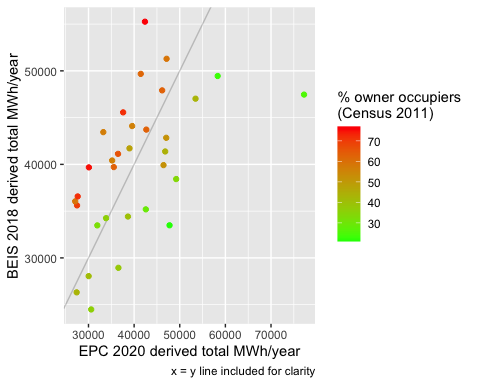
As we would expect those MSOAs with the lowest EPC coverage on both baseline measures tend to have higher proportions of owner occupiers.

We can use the same approach to compare estimates of total energy demand at the MSOA level. To do this we compare:

* estimated total energy demand in MWh/year derived from the EPC estimates. This energy only relates to current primary energy (space heating, hot water and lighting) and of course also suffers from missing EPCs (see above)
* observed electricity and gas demand collated by BEIS for their sub-national statistical series. This applies to all domestic energy demand but the most recent data is for 2018 so will suffer from the absence of dwellings that are present in the most recent EPC data (see above).

We should therefore not expect the values to match but we might reasonably expect a correlation.

ggplot2::ggplot(t, aes(x = sumEpcMWh,   
 y = beisEnergyMWh,  
 colour = round(ownerOcc\_pc))) +  
 geom\_abline(alpha = 0.2, slope=1, intercept=0) +  
 geom\_point() +  
 scale\_color\_continuous(name = "% owner occupiers \n(Census 2011)", high = "red", low = "green") +  
 #theme(legend.position = "bottom") +  
 labs(x = "EPC 2020 derived total MWh/year",  
 y = "BEIS 2018 derived total MWh/year",  
 caption = "x = y line included for clarity")



Energy demand comparison

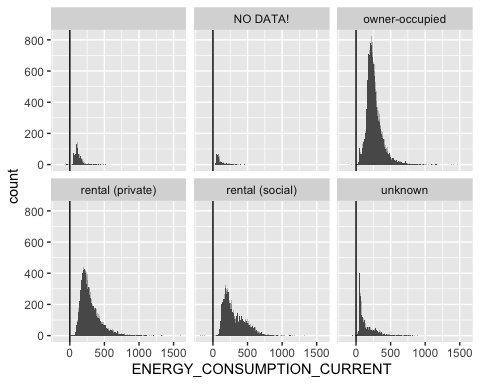
outlier <- t[sumEpcMWh > 70000]

@ref(fig:energyMSOAPlot) shows that both of these are true. MSOAs with a high proportion of owner occupiers (and therefore more likely to have missing EPCs) tend to have higher observed energy demand than the EOC data suggests - they are above the reference line. MSOAs with a lower proportion of owner occupiers (and therefore more likely to have more complete EPC coverage) tend to be on or below the line. As before we have the same notable outlier (E02003577) and for the same reasons… In this case this produces a much higher energy demand estimate than the BEIS 2018 data records

## Check ENERGY\_CONSUMPTION\_CURRENT

We recode the current energy consumption into categories for comparison with other low values and the presence of wind turbines/PV. We use -ve, 0 and 1 kWh as the thresholds of interest.

ggplot2::ggplot(sotonUniqueEPCsDT, aes(x = ENERGY\_CONSUMPTION\_CURRENT)) +  
 geom\_histogram(binwidth = 5) +   
 facet\_wrap(~TENURE) +  
 geom\_vline(xintercept = 0)



Histogram of ENERGY\_CONSUMPTION\_CURRENT

underZero <- nrow(sotonUniqueEPCsDT[ENERGY\_CONSUMPTION\_CURRENT < 0])  
  
t <- with(sotonUniqueEPCsDT[ENERGY\_CONSUMPTION\_CURRENT < 0],  
 table(BUILT\_FORM,TENURE))  
  
  
kableExtra::kable(t, caption = "Properties with ENERGY\_CONSUMPTION\_CURRENT < 0")

Properties with ENERGY\_CONSUMPTION\_CURRENT < 0

owner-occupied

rental (social)

unknown

Detached

0

2

0

0

End-Terrace

2

0

2

0

Mid-Terrace

3

1

1

0

NO DATA!

0

0

0

2

Semi-Detached

6

0

0

1

# do we think this is caused by solar/wind?  
sotonUniqueEPCsDT[, hasWind := ifelse(WIND\_TURBINE\_COUNT > 0, "Yes", "No")]  
#table(sotonUniqueEPCsDT$hasWind)  
sotonUniqueEPCsDT[, hasPV := ifelse(PHOTO\_SUPPLY >0, "Yes", "No")]  
#table(sotonUniqueEPCsDT$hasPV)  
sotonUniqueEPCsDT[, consFlag := ifelse(ENERGY\_CONSUMPTION\_CURRENT < 0, "-ve kWh/y", NA)]  
sotonUniqueEPCsDT[, consFlag := ifelse(ENERGY\_CONSUMPTION\_CURRENT == 0, "0 kWh/y", consFlag)]  
sotonUniqueEPCsDT[, consFlag := ifelse(ENERGY\_CONSUMPTION\_CURRENT > 0 &   
 ENERGY\_CONSUMPTION\_CURRENT <= 1, "0-1 kWh/y", consFlag)]  
sotonUniqueEPCsDT[, consFlag := ifelse(ENERGY\_CONSUMPTION\_CURRENT > 1, "1+ kWh/y", consFlag)]  
  
t <- sotonUniqueEPCsDT[, .(nObs = .N), keyby = .(consFlag, hasWind, hasPV)]  
  
kableExtra::kable(t, caption = "Properties in ENERGY\_CONSUMPTION\_CURRENT category by presence of microgeneration")

Properties in ENERGY\_CONSUMPTION\_CURRENT category by presence of microgeneration

consFlag

hasWind

hasPV

nObs

-ve kWh/y

NA

NA

5

-ve kWh/y

No

NA

15

0 kWh/y

NA

NA

1

0 kWh/y

No

No

1

1+ kWh/y

NA

NA

5550

1+ kWh/y

No

NA

33018

1+ kWh/y

No

No

32529

1+ kWh/y

No

Yes

447

1+ kWh/y

Yes

NA

6

1+ kWh/y

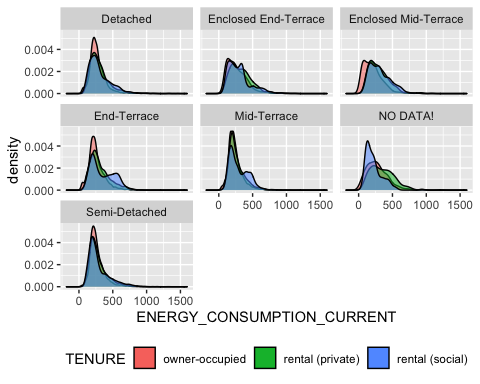
Yes

No

28

There are only 20 dwellings where ENERGY\_CONSUMPTION\_CURRENT < 0 and none of them seem to have PV or a wind turbine so we can probably ignore them.

# repeat with a density plot to allow easy overlap   
# exclude those with no data  
ggplot2::ggplot(sotonUniqueEPCsDT[TENURE != "NO DATA!" &  
 TENURE != "unknown" &  
 TENURE != ""], aes(x = ENERGY\_CONSUMPTION\_CURRENT,   
 fill = TENURE, alpha = 0.2)) +  
 geom\_density() +  
 facet\_wrap(~BUILT\_FORM) +  
 guides(alpha = FALSE) +  
 theme(legend.position = "bottom")

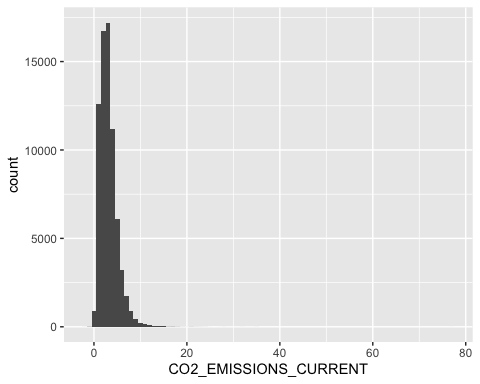


Comparing distributions of ENERGY\_CONSUMPTION\_CURRENT by tenure and built form

## Check CO2\_EMISSIONS\_CURRENT

Next we do the same for current emissions. Repeat the coding for total floor area using 0 and 1 TCO2/y as the threshold of interest.

ggplot2::ggplot(sotonUniqueEPCsDT, aes(x = CO2\_EMISSIONS\_CURRENT)) +  
 geom\_histogram(binwidth = 1)



Histogram of CO2\_EMISSIONS\_CURRENT

nZeroEmissions <- nrow(sotonUniqueEPCsDT[CO2\_EMISSIONS\_CURRENT < 0])  
  
sotonUniqueEPCsDT[, emissionsFlag := ifelse(CO2\_EMISSIONS\_CURRENT < 0, "-ve CO2/y", NA)]  
sotonUniqueEPCsDT[, emissionsFlag := ifelse(CO2\_EMISSIONS\_CURRENT == 0, "0 CO2/y", emissionsFlag)]  
sotonUniqueEPCsDT[, emissionsFlag := ifelse(CO2\_EMISSIONS\_CURRENT > 0 &   
 CO2\_EMISSIONS\_CURRENT <= 1, "0-1 TCO2/y", emissionsFlag)]  
sotonUniqueEPCsDT[, emissionsFlag := ifelse(CO2\_EMISSIONS\_CURRENT > 1, "1+ TCO2/y", emissionsFlag)]  
  
t <- sotonUniqueEPCsDT[, .(nObs = .N), keyby = .(emissionsFlag, hasWind, hasPV)]  
  
kableExtra::kable(t, caption = "Properties with CO2\_EMISSIONS\_CURRENT < 0 by presence of microgeneration")

Properties with CO2\_EMISSIONS\_CURRENT < 0 by presence of microgeneration

emissionsFlag

hasWind

hasPV

nObs

-ve CO2/y

NA

NA

4

-ve CO2/y

No

NA

16

-ve CO2/y

No

No

2

0 CO2/y

NA

NA

5

0 CO2/y

No

No

1

0-1 TCO2/y

NA

NA

3446

0-1 TCO2/y

No

NA

1916

0-1 TCO2/y

No

No

533

0-1 TCO2/y

No

Yes

19

0-1 TCO2/y

Yes

NA

1

0-1 TCO2/y

Yes

No

1

1+ TCO2/y

NA

NA

2101

1+ TCO2/y

No

NA

31101

1+ TCO2/y

No

No

31994

1+ TCO2/y

No

Yes

428

1+ TCO2/y

Yes

NA

5

1+ TCO2/y

Yes

No

27

kableExtra::kable(round(100\*(prop.table(table(sotonUniqueEPCsDT$emissionsFlag,   
 sotonUniqueEPCsDT$consFlag,   
 useNA = "always")  
 )  
 )  
 ,2)  
 , caption = "% properties in CO2\_EMISSIONS\_CURRENT categories by ENERGY\_CONSUMPTION\_CURRENT categories")

% properties in CO2\_EMISSIONS\_CURRENT categories by ENERGY\_CONSUMPTION\_CURRENT categories

-ve kWh/y

0 kWh/y

1+ kWh/y

NA

-ve CO2/y

0.03

0

0.00

0

0 CO2/y

0.00

0

0.00

0

0-1 TCO2/y

0.00

0

8.26

0

1+ TCO2/y

0.00

0

91.70

0

NA

0.00

0

0.00

0

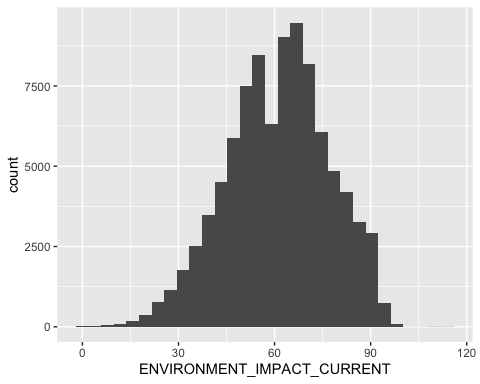
There are 22 properties with 0 or negative emissions. It looks like they are also the properties with -ve kWh as we might expect. So we can safely ignore them.

## Check ENVIRONMENT\_IMPACT\_CURRENT

Environmental impact should decrease as emissions increase.

ggplot2::ggplot(allEPCs\_DT, aes(x = ENVIRONMENT\_IMPACT\_CURRENT)) +  
 geom\_histogram()

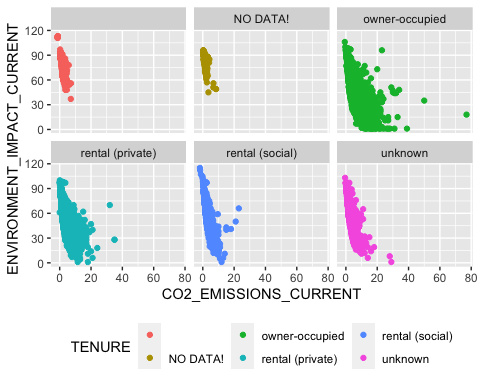
## `stat\_bin()` using `bins = 30`. Pick better value with `binwidth`.



Histogram of ENVIRONMENT\_IMPACT\_CURRENT

So what is the relationship between ENVIRONMENT\_IMPACT\_CURRENT and CO2\_EMISSIONS\_CURRENT? It is not linear… (Figure @ref(fig:checkEmissionsImpact)) and there are some interesting outliers.

ggplot2::ggplot(allEPCs\_DT, aes(x = CO2\_EMISSIONS\_CURRENT,   
 y = ENVIRONMENT\_IMPACT\_CURRENT,  
 colour = TENURE)) +  
 geom\_point() +  
 facet\_wrap(TENURE~.) +  
 theme(legend.position = "bottom")

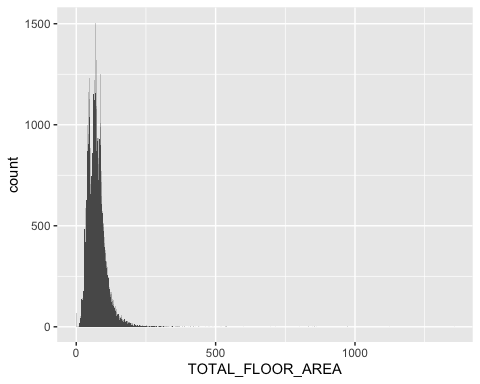


PLot of ENVIRONMENT\_IMPACT\_CURRENT vs CO2\_EMISSIONS\_CURRENT

## Check TOTAL\_FLOOR\_AREA

Repeat the coding for total floor area using 5 m2 as the threshold of interest.

ggplot2::ggplot(sotonUniqueEPCsDT, aes(x = TOTAL\_FLOOR\_AREA)) +  
 geom\_histogram(binwidth = 1)



Histogram of TOTAL\_FLOOR\_AREA

nZeroFloorArea <- nrow(sotonUniqueEPCsDT[TOTAL\_FLOOR\_AREA < 0])  
  
sotonUniqueEPCsDT[, floorFlag := ifelse(TOTAL\_FLOOR\_AREA == 0, "0 m2", NA)]  
sotonUniqueEPCsDT[, floorFlag := ifelse(TOTAL\_FLOOR\_AREA > 0 &   
 TOTAL\_FLOOR\_AREA <= 10, "0-5 m2", floorFlag)]  
sotonUniqueEPCsDT[, floorFlag := ifelse(TOTAL\_FLOOR\_AREA > 10, "5+ m2", floorFlag)]  
  
t <- with(sotonUniqueEPCsDT, table(floorFlag, consFlag))  
  
kableExtra::kable(round(100\*prop.table(t),2), caption = "% properties with TOTAL\_FLOOR\_AREA category by ENERGY\_CONSUMPTION\_CURRENT category")

% properties with TOTAL\_FLOOR\_AREA category by ENERGY\_CONSUMPTION\_CURRENT category

-ve kWh/y

0 kWh/y

1+ kWh/y

0 m2

0.00

0

0.10

0-5 m2

0.00

0

0.02

5+ m2

0.03

0

99.85

kableExtra::kable(head(sotonUniqueEPCsDT[, .(BUILDING\_REFERENCE\_NUMBER, PROPERTY\_TYPE, TOTAL\_FLOOR\_AREA,   
 ENERGY\_CONSUMPTION\_CURRENT)][order(-TOTAL\_FLOOR\_AREA)], 10),   
 caption = "Top 10 by floor area (largest)")

Top 10 by floor area (largest)

BUILDING\_REFERENCE\_NUMBER

PROPERTY\_TYPE

TOTAL\_FLOOR\_AREA

ENERGY\_CONSUMPTION\_CURRENT

4.697565e-314

House

1353.680

140

1.894551e-314

House

1123.000

120

4.846111e-314

House

973.210

522

2.559778e-314

House

861.360

279

8.172097e-315

House

855.000

170

4.440838e-315

House

846.421

161

1.933817e-314

House

833.000

206

4.076249e-314

House

800.000

185

1.280057e-314

House

714.000

224

2.444460e-314

Flat

694.000

88

kableExtra::kable(head(sotonUniqueEPCsDT[, .(BUILDING\_REFERENCE\_NUMBER, PROPERTY\_TYPE, TOTAL\_FLOOR\_AREA,  
 ENERGY\_CONSUMPTION\_CURRENT)][order(TOTAL\_FLOOR\_AREA)], 10),   
 caption = "Bottom 10 by floor area (smallest)")

Bottom 10 by floor area (smallest)

BUILDING\_REFERENCE\_NUMBER

PROPERTY\_TYPE

TOTAL\_FLOOR\_AREA

ENERGY\_CONSUMPTION\_CURRENT

9.111592e-316

Flat

0

104

3.102124e-315

Flat

0

58

3.294384e-315

Flat

0

110

3.695003e-315

Flat

0

70

4.369619e-315

Flat

0

119

4.371685e-315

Flat

0

71

7.302298e-315

Flat

0

115

9.515727e-315

Flat

0

144

9.562249e-315

Flat

0

103

1.059979e-314

Flat

0

117

kableExtra::kable(round(100\*prop.table(t),2), caption = "% properties with TOTAL\_FLOOR\_AREA category by ENERGY\_CONSUMPTION\_CURRENT category")

% properties with TOTAL\_FLOOR\_AREA category by ENERGY\_CONSUMPTION\_CURRENT category

-ve kWh/y

0 kWh/y

1+ kWh/y

0 m2

0.00

0

0.10

0-5 m2

0.00

0

0.02

5+ m2

0.03

0

99.85

@ref(tab:checkEmissions) shows that the properties with floor area of < 10m2 are not necessarily the ones with 0 or negative kWh values. Nevertheless they represent a small proportion of all properties.

The scale of the x axis also suggests a few very large properties.

## Data summary

We have identified some issues with a small number of the properties in the EPC dataset. These are not unexpected given that much of the estimates rely on partial or presumed data. Data entry errors are also quite likely. As a result we exclude:

* any property where ENERGY\_CONSUMPTION\_CURRENT <= 0
* any property where TOTAL\_FLOOR\_AREA <= 5
* any property where CO2\_EMISSIONS\_CURRENT <= 0

finalDT <- sotonUniqueEPCsDT[ENERGY\_CONSUMPTION\_CURRENT > 0 &  
 TOTAL\_FLOOR\_AREA > 5 &  
 CO2\_EMISSIONS\_CURRENT > 0]  
  
skimr::skim(finalDT)

## Warning: Couldn't find skimmers for class: integer64; No user-defined `sfl` provided. Falling back to `character`.

Data summary

Name

finalDT

Number of rows

71501

Number of columns

23

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Column type frequency:

character

12

Date

1

numeric

10

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Group variables

None

**Variable type: character**

skim\_variable

n\_missing

complete\_rate

min

max

empty

n\_unique

whitespace

BUILDING\_REFERENCE\_NUMBER

0

1.00

17

21

0

71501

0

LMK\_KEY

0

1.00

29

34

0

71501

0

PROPERTY\_TYPE

0

1.00

4

10

0

5

0

BUILT\_FORM

0

1.00

8

20

0

7

0

TENURE

0

1.00

0

16

1906

6

0

POSTCODE

0

1.00

8

8

0

5105

0

LOCAL\_AUTHORITY\_LABEL

0

1.00

11

11

0

1

0

hasWind

5547

0.92

2

3

0

2

0

hasPV

38499

0.46

2

3

0

2

0

consFlag

0

1.00

8

8

0

1

0

emissionsFlag

0

1.00

9

10

0

2

0

floorFlag

0

1.00

5

6

0

2

0

**Variable type: Date**

skim\_variable

n\_missing

complete\_rate

min

max

median

n\_unique

LODGEMENT\_DATE

0

1

2008-10-01

2020-06-30

2014-10-22

4132

**Variable type: numeric**

skim\_variable

n\_missing

complete\_rate

mean

sd

p0

p25

p50

p75

p100

hist

ENVIRONMENT\_IMPACT\_CURRENT

0

1.00

62.52

15.70

1.00

52.0

63.0

73

100.00

▁▂▆▇▂

ENERGY\_CONSUMPTION\_CURRENT

0

1.00

263.17

140.44

4.00

174.0

233.0

327

1597.00

▇▂▁▁▁

CO2\_EMISSIONS\_CURRENT

0

1.00

3.16

1.94

0.10

1.8

2.8

4

77.00

▇▁▁▁▁

PHOTO\_SUPPLY

38499

0.46

0.59

5.12

0.00

0.0

0.0

0

100.00

▇▁▁▁▁

WIND\_TURBINE\_COUNT

5547

0.92

0.00

0.02

-1.00

0.0

0.0

0

1.00

▁▁▇▁▁

TOTAL\_FLOOR\_AREA

0

1.00

73.05

34.87

5.85

49.0

69.0

87

1353.68

▇▁▁▁▁

epcIsSocialRent

0

1.00

0.21

0.40

0.00

0.0

0.0

0

1.00

▇▁▁▁▂

epcIsPrivateRent

0

1.00

0.27

0.44

0.00

0.0

0.0

1

1.00

▇▁▁▁▃

epcIsOwnerOcc

0

1.00

0.41

0.49

0.00

0.0

0.0

1

1.00

▇▁▁▁▆

epcIsUnknownTenure

0

1.00

0.04

0.19

0.00

0.0

0.0

0

1.00

▇▁▁▁▁

This leaves us with a total of 71,501 properties. ` # Current estimated annual CO2 emmisions

We can now use the cleaned data to estimated the annual CO2 emissions at:

* MSOA level for Southampton using
  + BEIS observed data
  + aggregated EPC data
* Dwelling level for Southampton using
  + aggregated EPC data

Obviously the EPC-derived totals will not be the total CO2 emissions for **all** Southampton properties since we know not all dwellings are represented in the EPC data (see above).

## MSOA estimates

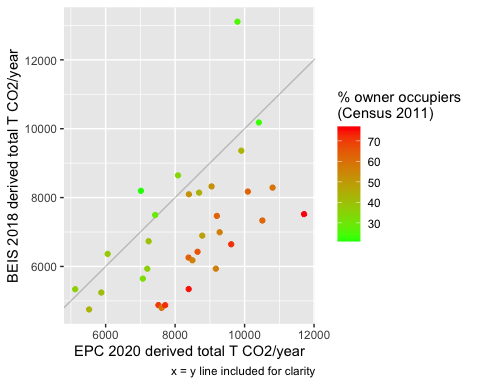
Method:

elecCF <- 200 # CO2e/kWh https://www.icax.co.uk/Grid\_Carbon\_Factors.html  
gasCF <- 215 # https://www.icax.co.uk/Carbon\_Emissions\_Calculator.html

BEIS: apply 2019 mean grid carbon intensity for: \* electricity: 200 g \* gas: 215 g CO2e/kWh EPC: use estimated CO2 values - note based on ‘old’ electricity grid carbon intensity values ()

sotonMSOA\_DT[, sumBEIS\_tCO2 := (beisElecMWh/1000)\*elecCF + (beisGasMWh/1000)\*gasCF]

ggplot2::ggplot(sotonMSOA\_DT, aes(x = sumBEIS\_tCO2,   
 y = sumEPC\_tCO2,  
 colour = round(ownerOcc\_pc))) +  
 geom\_abline(alpha = 0.2, slope=1, intercept=0) +  
 geom\_point() +  
 scale\_color\_continuous(name = "% owner occupiers \n(Census 2011)", high = "red", low = "green") +  
 #theme(legend.position = "bottom") +  
 labs(x = "EPC 2020 derived total T CO2/year",  
 y = "BEIS 2018 derived total T CO2/year",  
 caption = "x = y line included for clarity")



Energy demand comparison

#outlier <- t[sumEpcMWh > 70000]

@ref(fig:energyMSOAPlot) shows that

```

Scenarios

National Grid’s Future Energy Scenarios:

* 2030 emissions level for electricity of 0.102 kgCO2/kWh
* gas unchanged

# R packages used

* rmarkdown (Allaire et al. 2018)
* bookdown (Xie 2016a)
* knitr (Xie 2016b)
* data.table (Dowle et al. 2015)
* ggplot2 (Wickham 2009)
* kableExtra (Zhu 2018)
* readxl (Wickham and Bryan 2017)

# References

Allaire, JJ, Yihui Xie, Jonathan McPherson, Javier Luraschi, Kevin Ushey, Aron Atkins, Hadley Wickham, Joe Cheng, and Winston Chang. 2018. *Rmarkdown: Dynamic Documents for R*. <https://CRAN.R-project.org/package=rmarkdown>.

Dowle, M, A Srinivasan, T Short, S Lianoglou with contributions from R Saporta, and E Antonyan. 2015. *Data.table: Extension of Data.frame*. <https://CRAN.R-project.org/package=data.table>.

Wickham, Hadley. 2009. *Ggplot2: Elegant Graphics for Data Analysis*. Springer-Verlag New York. <http://ggplot2.org>.

Wickham, Hadley, and Jennifer Bryan. 2017. *Readxl: Read Excel Files*. <https://CRAN.R-project.org/package=readxl>.

Xie, Yihui. 2016a. *Bookdown: Authoring Books and Technical Documents with R Markdown*. Boca Raton, Florida: Chapman; Hall/CRC. <https://github.com/rstudio/bookdown>.

———. 2016b. *Knitr: A General-Purpose Package for Dynamic Report Generation in R*. <https://CRAN.R-project.org/package=knitr>.

Zhu, Hao. 2018. *KableExtra: Construct Complex Table with ’Kable’ and Pipe Syntax*. <https://CRAN.R-project.org/package=kableExtra>.