**Tech Conferences – Speaker’s Notes**

**Slide 1**

Thanks Oskar. So I’m Nigel and for those I’ve not met, I used to teach maths and computer science, but I’m transitioning into data science. Over the last three years, I’ve dipped my toe into the conference circuit and I’d like to encourage more of you to go to conferences. Then we’ll finish off with some Machine Learning (don’t worry, I do realise half of you have other roles in data, so I’ll explain the process in simple terms.)

**[click]**

**Slide 2**

**[click]** In 2019, I travelled to Warsaw for PyCon PL

Next, I went to **[click]** Paris, for dotPy

Conferences didn’t happen in person for a couple of years, but

This summer I went to Py Data, **[click]** in London

And Euro Python **[click]** in Dublin

I love maps, but I reckon you just want to see…

**[click]**

**Slide 3**

The dataset!

**[click]**

Ooh, rows and columns. We’re in our comfort zone.

As you can see, I put together a few features, and I’d just like to highlight a couple of figures:

* Yes, it really did take me 120 hours to travel to Warsaw. I don’t fly, because of climate change, so I made a little roadtrip of it and stopped off in Prague and Budapest.
* Euro Python offered *education* discount, for teachers as well as students

Then I did some lighthearted ratings **[click]**

* So the takeaway message is: they’re all really friendly. A few people did seem to be with colleagues, but most were on their own and I had some lovely chats.
* The vegan catering was disappointing but I didn’t starve
* They weren’t all straight, white men. Five start for Euro Python, who had a pre-conference training day for trans coders
* I got some great swag. One the last day, one company was practically begging me to take several sets of bike bike lights. Poland was disappointing, though: I didn’t even get a T shirt
* The talks were all good. All the conferences flagged which talks were beginner friendly
* And the venues were all comfortable and modern

Now, have any eagle eyed data detectives spotted anything unusual? It’s ok, we’re not at school, it’s ok to shout out **[wait]**

Yeah, there’s no data for Paris. Any idea why? **[wait]**

March 2020. I got as far as Paris, had a nice weekend visiting friends…then they cancelled the conference because of some new virus. So, I’ll never know what I missed in Paris.

…Except, with our secret superpowers, let’s find out!

**[click]**

**Slide 4 [click]**

So with any Machine Learning model, the first thing you do is to split the data. **[click]**

You use about 2/3 for training. The machine looks for patterns, then formulates rules **[click]**

Then you use those rules to make predictions for the unseen test data **[click]**

Then you see how close your predictions were to the actual values **[click]**

Then you can use your model to predict new data, and say how confident you are in your predictions **[click]**

**Slide 5**

Now, for those of you who’ve never done linear regression, yes you have. It’s just scattergraphs. Well, scattergraphs on steroids. You remember the process: **[click]**

1. You plot your data **[click]**
2. Then you draw a line of best fit (aka the regression line) **[click]**
3. Then you use the line to make predictions. Just like GCSE. **[click]**

So we could use the number of participants to predict the friendliness. Our machine has randomly selected 2/3 of the data for training **[click]**

So we plot the two datapoints: 250 people is 4-star friendly, and 400 people is 5-star friendly

Then our swarm of dots is just two, so we just join them up. With a straight line. Using a ruler.

*Ruler?* [groan]

*Lend us yer fag packet*

*If he takes it you’re dead*

Then we can predict the friendliness of the unseen test data **[click]**

300 people **[click]**

Should be **[click]**

4.33-star friendly **[click]**

And the actual value was 4, so I’m quite pleased with that accuracy **[click]**

**Slide 6**

Now that was an example of interpolation because 300 was within the limits of the data we trained on. **[click]**

**Slide 7**

If we used travel time to predict friendliness, we plot the points and draw the line again, then…

Oh dear

*Sir, can I just stick some more paper on the side?*

So you start again and do it properly. **[click]**

**Slide 8 [click]**

Then you borrow your mate’s fag packet again and extend the line…then extend it a bit more…

And you probably go a bit wonky, but it’s still accurate enough for GCSE

So you make your prediction **[click]** from 120 **[click]**

Down to the line **[click]**

Then read the height: minus 7

I don’t know about you, but I don’t think I’d like to go to a conference with minus 7 star friendliness

It’d be like taking my python-soul into a room full of R-sssspirits!

Right. That was using one feature to predict the height. If you use two features, you need… **[click]**

**Slide 9**

3D coordinates – with a regression plane.

Now, I’ve made it as simple as I can. I’ve kept everything on the axes and labelled the heights, but it’s still not that easy to visualise **[click]**

Looking down, we could focus on the z heights, in blue **[click]**

The formula for z is actually really simple: you just add x and y together **[click]**

I told you I love maps, so we could use contour lines to visualise it **[click] [click] [click] [click] [click]**

**Slide 10**

But it’s obviously the formula that the machine uses to make predictions: in this case 2+1 = 3 **[click]**

**Slide 11**

So if we’re using **[click]**

One feature, we need **[click]**

2D coordinates

If we’re using **[click]**

Two features, we need **[click]**

3D coordinates **[click]**

By extension, three features would need 4D coordinates – which I’m certainly not going to attempt to visualise! **[click]**

But the conferences model has seven features, so the machine is calculating with 8D coordinates!

But the idea is ‘just’ an extension of GCSE scattergraphs **[click]**

**Slide 12**

I’ll just finish off by showing you how little code you need to predict what I missed in Paris **[click]**

So that’s all the code, but to put it in perspective **[click]**

Most of it is just setting up: removing the bits we don’t want, then separating the features and ratings **[click]**

The actual model **[click]** is tiny! So I’ll quickly talk you through it **[click]**

**Slide 13**

First we split the data **[click]**

The next line sets up an instance of linear regression – a bit like, “New page. Copy the title and date. Get your rulers out.” **[click]**

Fitting the model means plotting the training data and drawing the regression line **[click]**

Then we can predict the test data **[click]**

And 3.74 is pretty close to the actual value of 4. Happy days

So, finally, we can gaze into our *boule de cristal* and see that I missed **[click]** a 3.73-star conference in Paris