Why the sample code initialize the Matrix wrongly (transposed), but still gives correct results?

Junsheng_Fu May '17

Hello all,

In Lesson 7 (Unscented Kalman Filters) 21. Sigma Point Prediction Assigment 2. For these following codes, I find it is very interesting.

In `main.cpp`,

```
MatrixXd Xsig_pred = MatrixXd(15, 5);  // line 20
```

In `ukf.cpp`

```
int n_x = 5; // line 24
int n_aug = 7; // line 27
MatrixXd Xsig_pred = MatrixXd(n_x, 2 * n_aug + 1); // line 41
*Xsig_out = Xsig_pred; // line 101:
```

Why is the last assignment valid? Because `*Xsig_out` and `Xsig_pred` have different dimensions.

To my surprise, if I change the line 20 of main.cpp as follow initialization, the code give me the same final output.

```
MatrixXd Xsig_pred = MatrixXd(5, 15);
```

> Solved by subodh.malgonde in post #4
>
> @Junsheng_Fu @Alex_Cui @fernandodamasio If you notice in ukf.cpp, a pointer is being passed as a function argument and not a reference: void UKF::GenerateSigmaPoints(MatrixXd* Xsig_out) { ..... other code ... //write result *Xsig_out = Xsig; At the end we just change the contents of…

fernandodamasio May '17

Hi @Junsheng_Fu,

You have a great eye!!! I can see the issue but I can't see why it's working, I agree with you that there is something wrong… I need to hear more opinions.

Maybe it's a bug in the code but the backend of the Udacity platform is working.

Let's wait for more comments.

Regards,

Alex_Cui May '17

Hi @Junsheng_Fu,

It seems the dimension definition in the main is wrong. It should be (5, 15). Can you file a bug report on waffle?

subodh.malgonde May '17

@Junsheng_Fu @Alex_Cui @fernandodamasio
If you notice in `ukf.cpp`, a pointer is being passed as a function argument and not a reference:

```
void UKF::GenerateSigmaPoints(MatrixXd* Xsig_out) {
     ..... other code ...

    //write result
    *Xsig_out = Xsig;
```

At the end we just change the contents of the memory location where the pointer to `Xsig_out` points to. I think its due to this that the variable initialization does not matter.

I haven't done C++ for a long time, so not sure if this explanation is correct 😄