# Individual specific effects in myotonic dystrophy and other diseases

Adam Kurkiewicz MVLS/EPSRC PhD studentship application

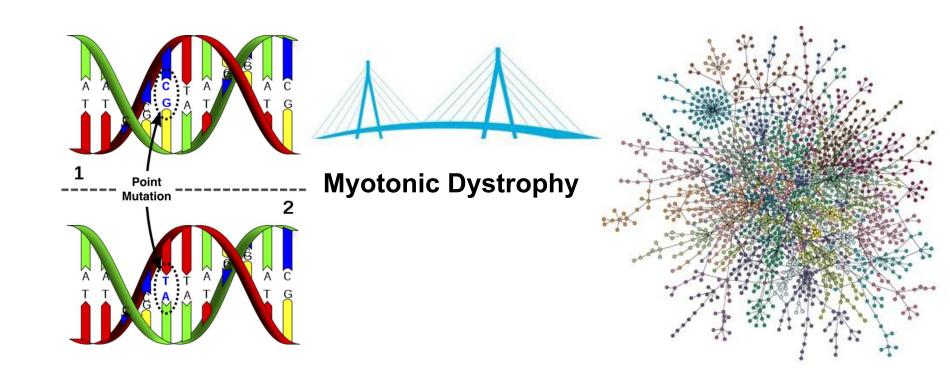
# Myotonic Dystrophy



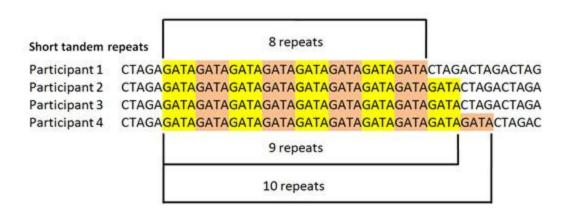


## Single-Gene Disease

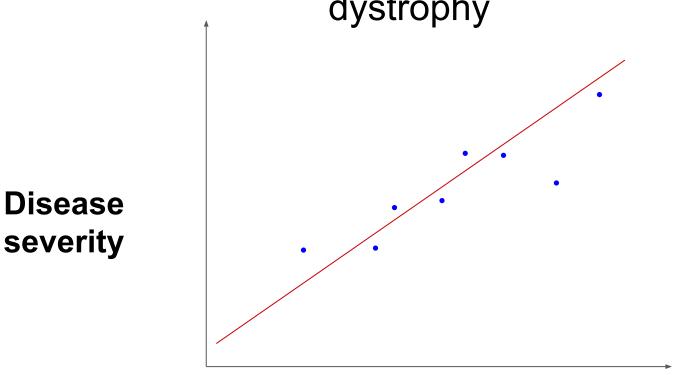
# **Complex Disease**



#### Short Tandem Repeats (STRs)

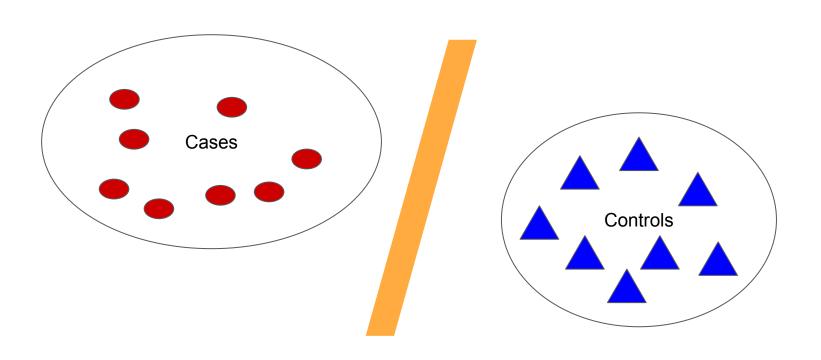


Genotype to phenotype correlations in myotonic dystrophy

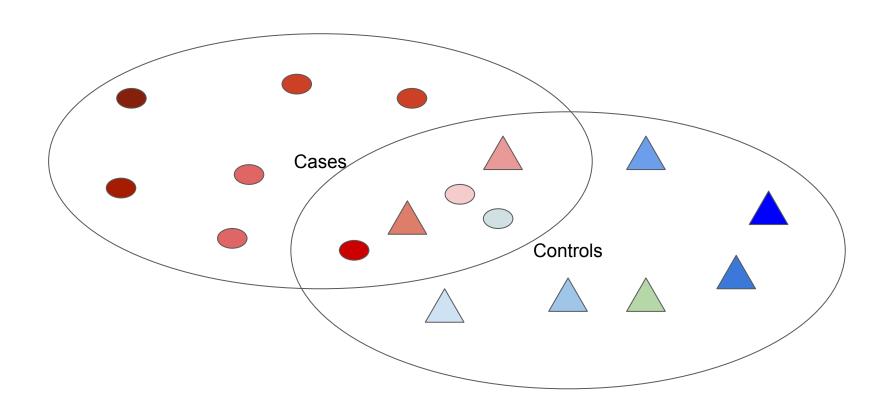


Length of STR

# Fallacy of Case / Control Paradigm

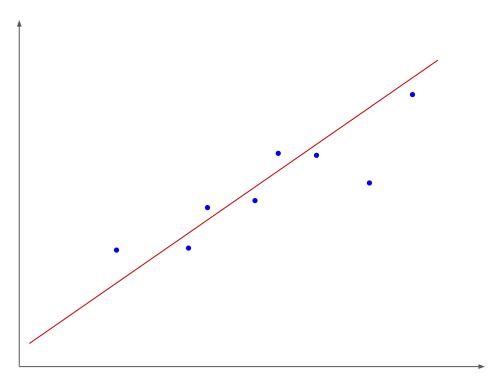


# Fallacy of Case / Control Paradigm



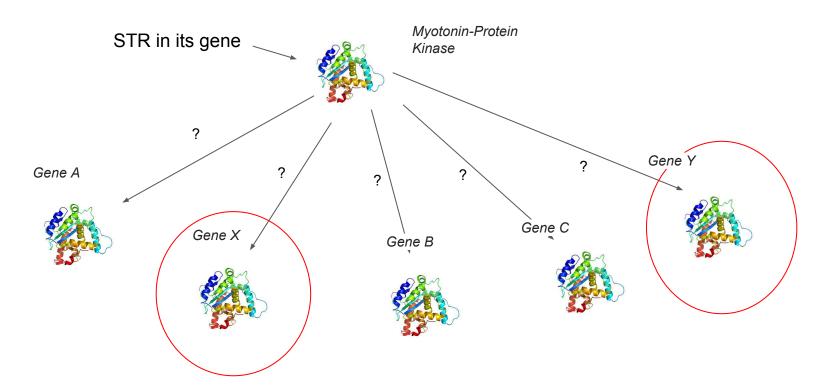
## Linear regression





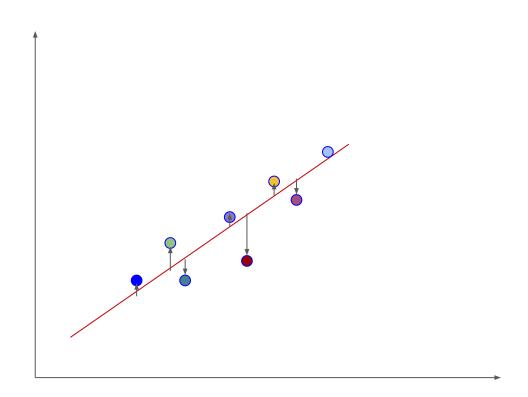
**Length of STR** 

# How many genes affected?



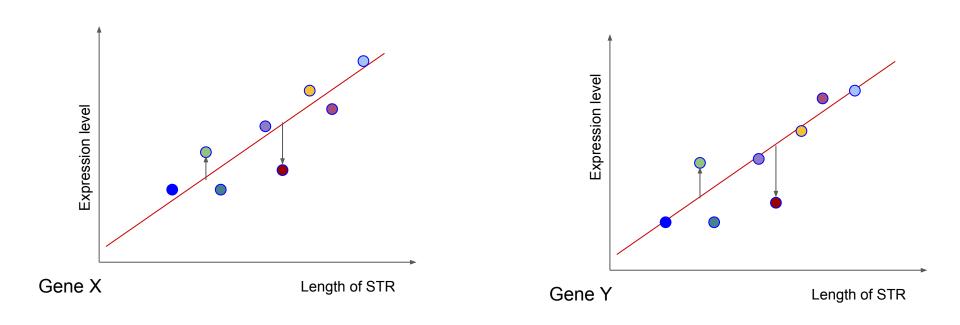
# **Transcriptomics**

Gene X expression level in tissue Y



Length of STR

#### Individual specific effects



The pattern seems to work for many genes at a time With highly significant deviations from expected for individuals

#### Research Questions for a PhD project

- 1. How can we use Individual-Specific Effects to normalise the data and better identify disease relevant genes and pathways?
- 2. How do they influence disease severity?
- 3. What is the cause of these Individual-Specific Effects?
- 4. How can we apply this approach to other diseases?
- How can we apply this approach to other omics datasets (e.g. proteomics, metabolomics)

#### Why me?

First Class programmer – all programming courses at A5 or higher.

Python – A4 (UoG)

Java – A4 (UoG)

C - A5 (UoG)

SAS - 94% (SAS Ltd.)

Also Javascript, Go, Haskell

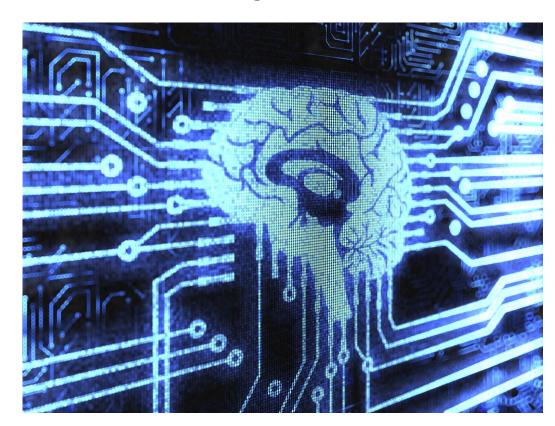
```
x Comultidime x O 2048wc/bc x 
                   B GitHub, Inc. [US] https://github.com/2048wc/2048wc/blob/master/boardLib/boardLib.go
🔛 Apps 🄌 exploit 💿 piotr 📳 important number 🐚 Formal proofs 🐚 skyscanner 🗐 todo 👢 lean 🐚 john's wisdom 🐚 recompile virtualb 📳 Wedding
                                             func (move *moveT) resolveBoardAtIndex(ism *iterationStateMachine) {
                                                                    giveHope := func() {
                                                                                             _, position := move.NewBoard.incrementFromBySteps(&move.OldBoard,
                                                                                                                      ism.currentIndex, ism.smallStepForward, ism.distance)
                                                                                              ism.mergeHopefulIndex = ism.currentIndex
                                                                                              ism.mergeHopefulDestination = position
                                                                                              ism.isMergeHopeful = true
                                                                                              if ism.distance == 0 {
                                                                                                                      ism.isHopefulUnmoved = true
                                                                                             } else {
                                                                                                                      ism.isHopefulUnmoved = false
                                                                     abandonHope := func() {
                                                                                              ism.isMergeHopeful = false
                                                                                              ism.isHopefulUnmoved = false
                                                                                              ism.mergeHopefulDestination = positionT{-1, -1}
                                                                                              ism.mergeHopefulIndex = positionT{-1, -1}
                                                                     dispatchLoser := func() {
                                                                                              if ism.isHopefulUnmoved == true {
                                                                                                                     move.NonMovedTiles = append(move.NonMovedTiles,
                                                                                                                                              ism.mergeHopefulIndex)
                                                                                             } else {
                                                                                                                      move.NonMergeMoves = append(move.NonMergeMoves,
                                                                                                                                              nonMergeMoveT{ism.mergeHopefulIndex,
                                                                                                                                                                      ism.mergeHopefulDestination})
```

# Machine Learning

Machine Learning often programmed in Python or Java

Project co-supervised by a Machine Learning specialist

First class result in Machine Learning.



#### Real, sustained interest in Natural Science

A named author of an article on bacterial membrane composition published in Journal of Analytical Chemistry [1]

A dissertation on the effect of evolution on relatedness at University of Glasgow [2]. Dissertation graded 1st class.

1st and 2nd year Biology, 1st year Chemistry

- 1. Kurkiewicz S, Kurkiewicz A. Profiling of bacterial cellular fatty acids by pyrolytic derivatization to 3-pyridylcarbinol esters. *Journal of Analytical Chemistry*. 2015;70(10):1225-1228.
- The effect of natural selection on relatedness in randomly mating population. https://github. com/picrin/naturalSelection/blob/master /dissertation/l4proj.pdf

#### Advanced understanding of maths and stats

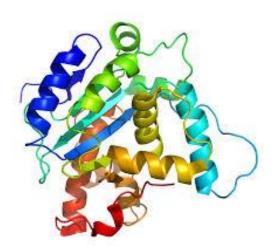
Joint Honours CS + Mathematics.

Courses in Algebra, Number Theory, Analysis, Topology, Galois Theory among others.

#### **Data Scientist**

⊞	Myotonic Dystrophy			
	ēra.	£ % .0 <sub>←</sub> .00 <sub>→</sub> 123 → Arial	- 10 - B / S	<u>A</u> . D H EE .
fx				
	А	В	С	D
1	STR Length	svc	PDS	UTV
2	16	79.30503154	53.66494735	99.39480806
3	2	27.60830944	49.5799717	95.54343672
4	6	39.05532438	58.25547738	71.37035929
5	75	79.71036365	29.36261104	48.92653332
6	8	51.31822568	61.10106565	74.75878487
7	56	96.25467951	61.91121989	1.472790237
8	56	35.49397726	2.659524191	80.56169642
9	86	53.92754623	59.69084768	15.64899263
10	27	61,52810613	85.95784659	41.01678275
11	11	1.138643908	37.37886323	84.13838703
12	34	44.07580252	24.64222558	42.90677395
13	47	13.27244813	56.86627767	58.34800339
14	98	0.3442644536	92.84082127	91.83154141
15	69	97.75043892	18.01711082	22.1369293
16	31	50.12865421	20.97989761	62.20895579
17	61	39.9197019	51.82645568	58.03116393
18	7	61.55727941	6.693946769	72.77012805
19	9	5.783592283	97.61782404	78.19065519
20	81	71.06162991	34.97044669	72.31466216
21	44	42.5900577	70.92045384	17.3941752
22	95	31.08689137	26.66984376	45.49154693
23	44	43.29324197	53.66714888	10.30762802
24	3	64.05856009	59.56142465	73.37129992
25	57	65.90402343	63.3533141	0.3014363331
26	64	26.12079261	31.01566545	8.574997453
27		00.74500045	00 70770077	50 00000174

#### Molecular Biologist



Project Title: Individual Specific Effects in Omics Datasets

Applicant: Adam Kurkiewicz

Key Strengths: Programming, Science, Statistics, Mathematics.

#### Me

