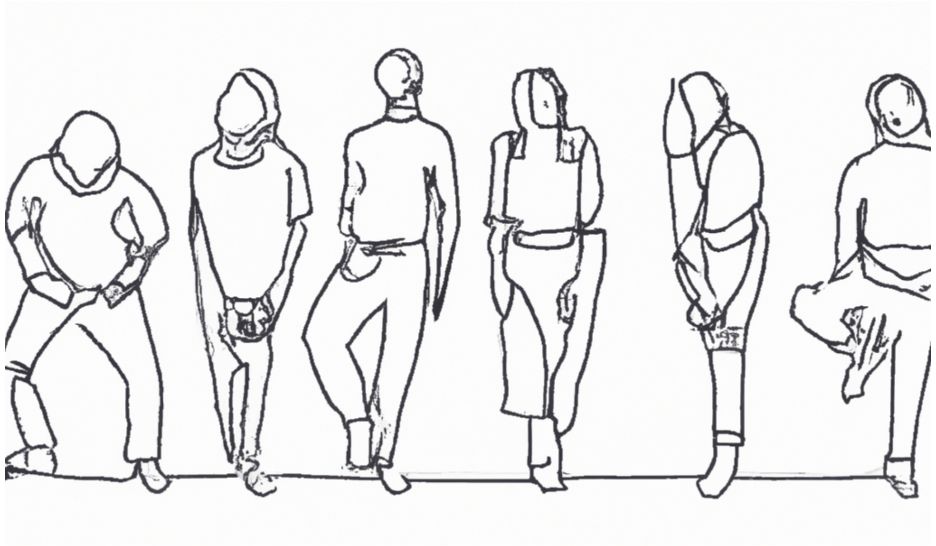
PS5210
Applied Neuroscience Methods

Workshop: analysis of behaviour



Analysis of behaviour

1. Participants perform a task, and we record their responses (multiple repetition since behaviour is variable)
2. <u>We calculate summary measures of individual performance</u>
3. We do statistical test to estimate effects and make inferences on population

Analysis of behaviour

1. Participants perform a task, and we record their responses (multiple repetition since behaviour is variable)
2. <u>We calculate summary measures of individual performance</u>
3. We do statistical test to estimate effects and make inferences on population

- Sometimes summary measures can be just averages (e.g., mean response time), but that is not always appropriate
- In many cases is useful to examine behaviour through the lens of a *mathematical model*

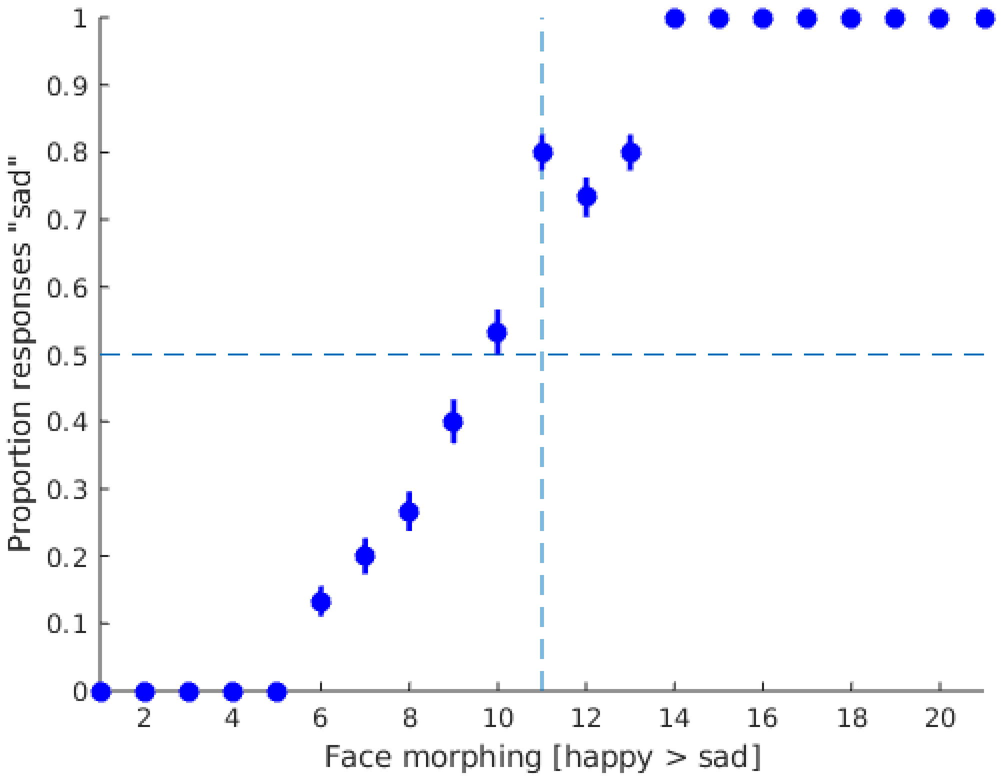
Example: emotion recognition task

- Participants are shown an image out of morphed continuum and indicate whether it is happy/sad

https://github.com/mattelisi/NeuroMethods/tree/master/experimental_tasks/emotion-recognition

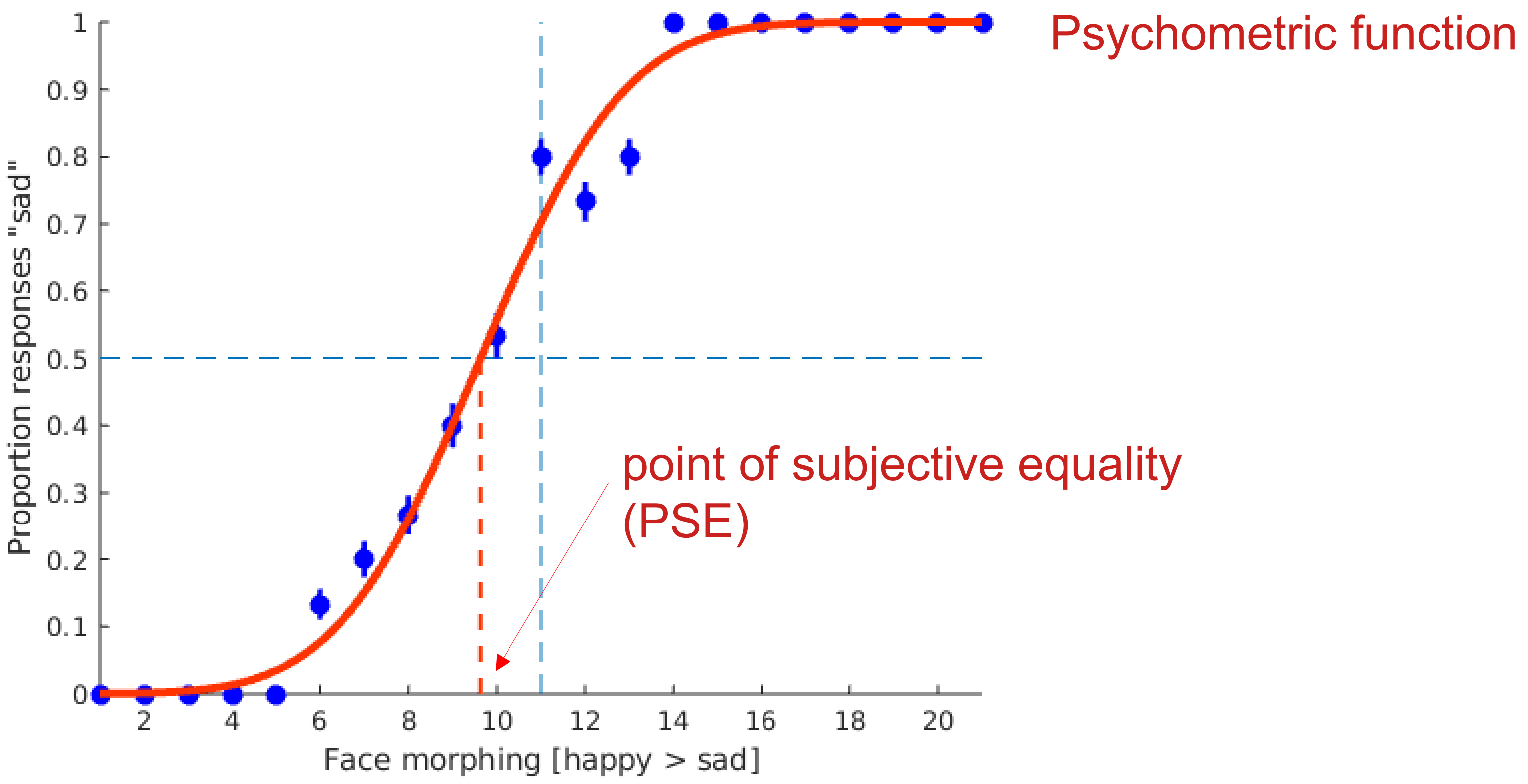
Example: emotion recognition task

- Example of data

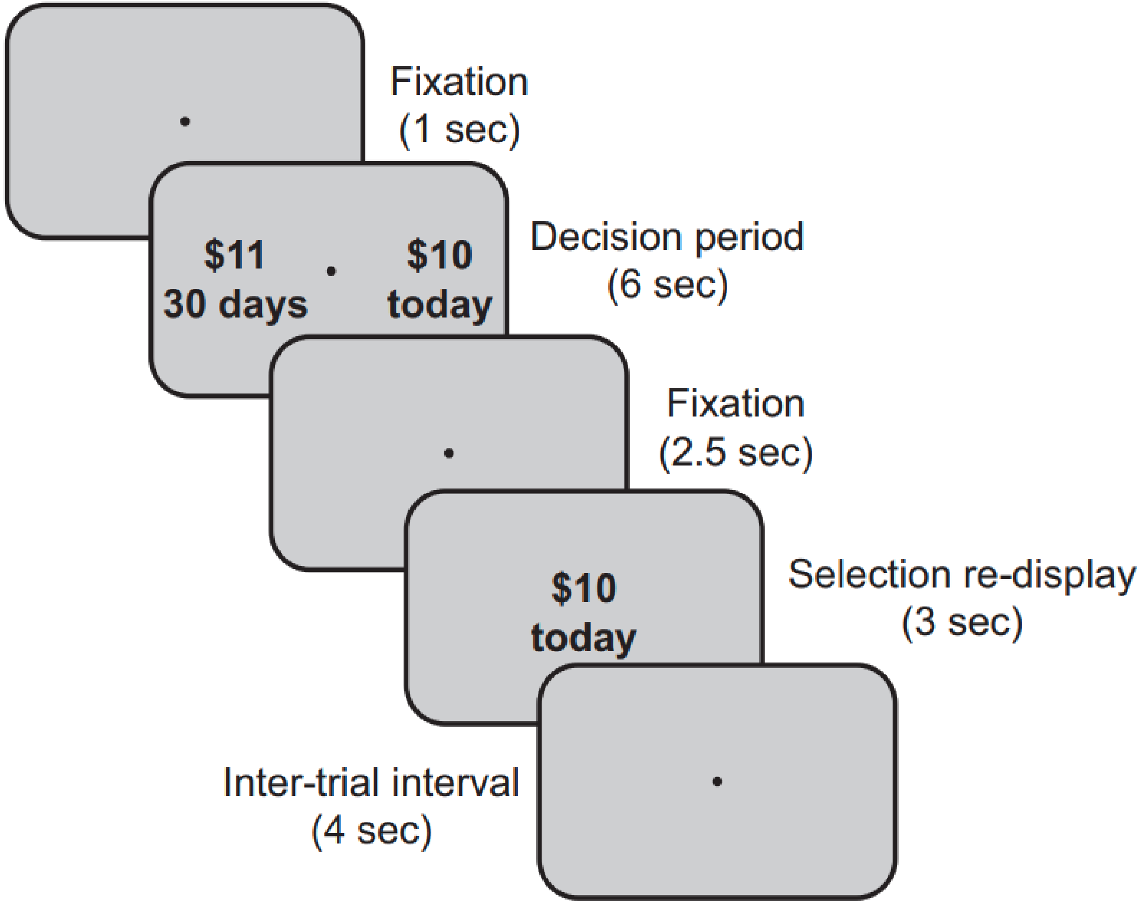


Example: emotion recognition task

- Example of data



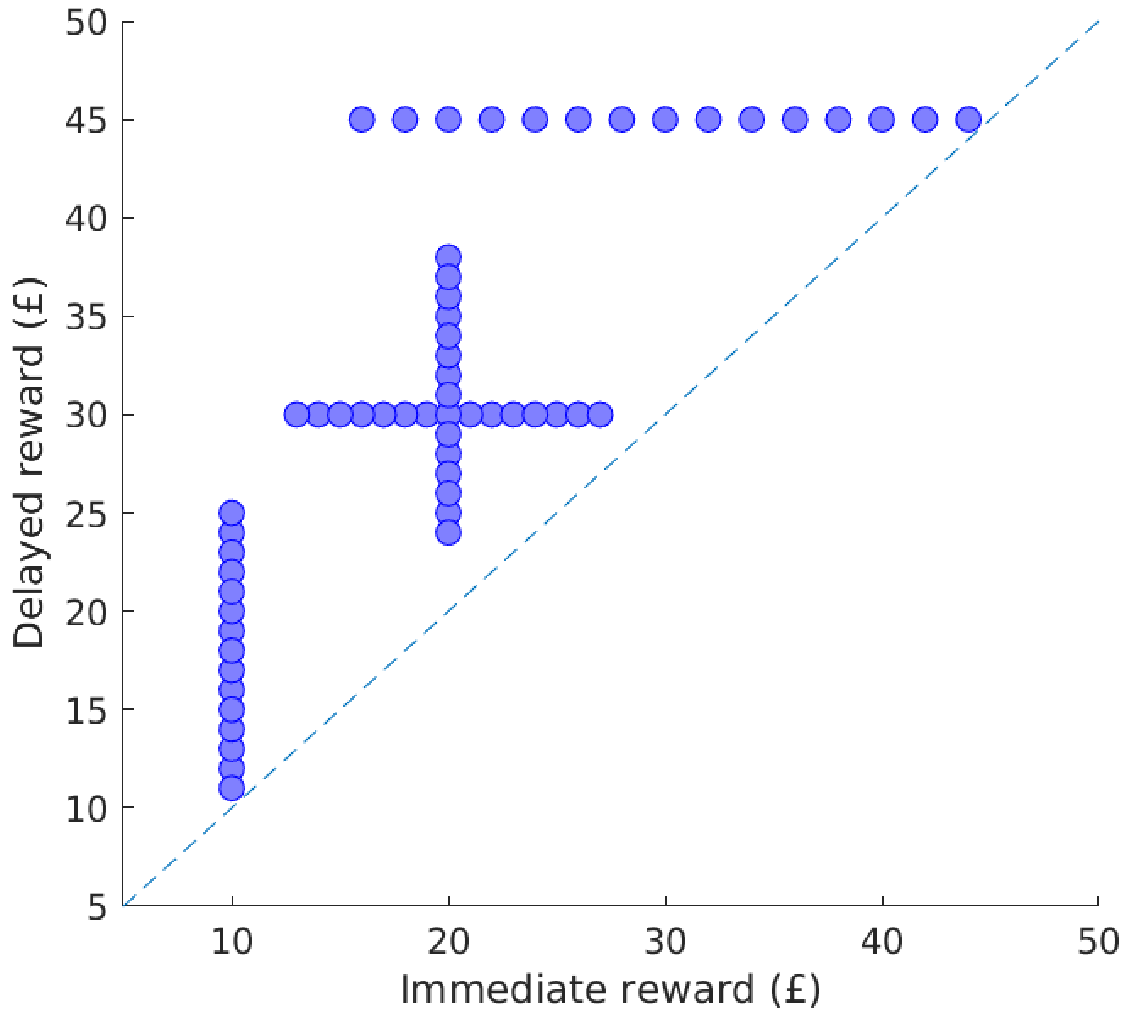
Example: temporal discounting task



Reppert et al, J Neurosci, 2015

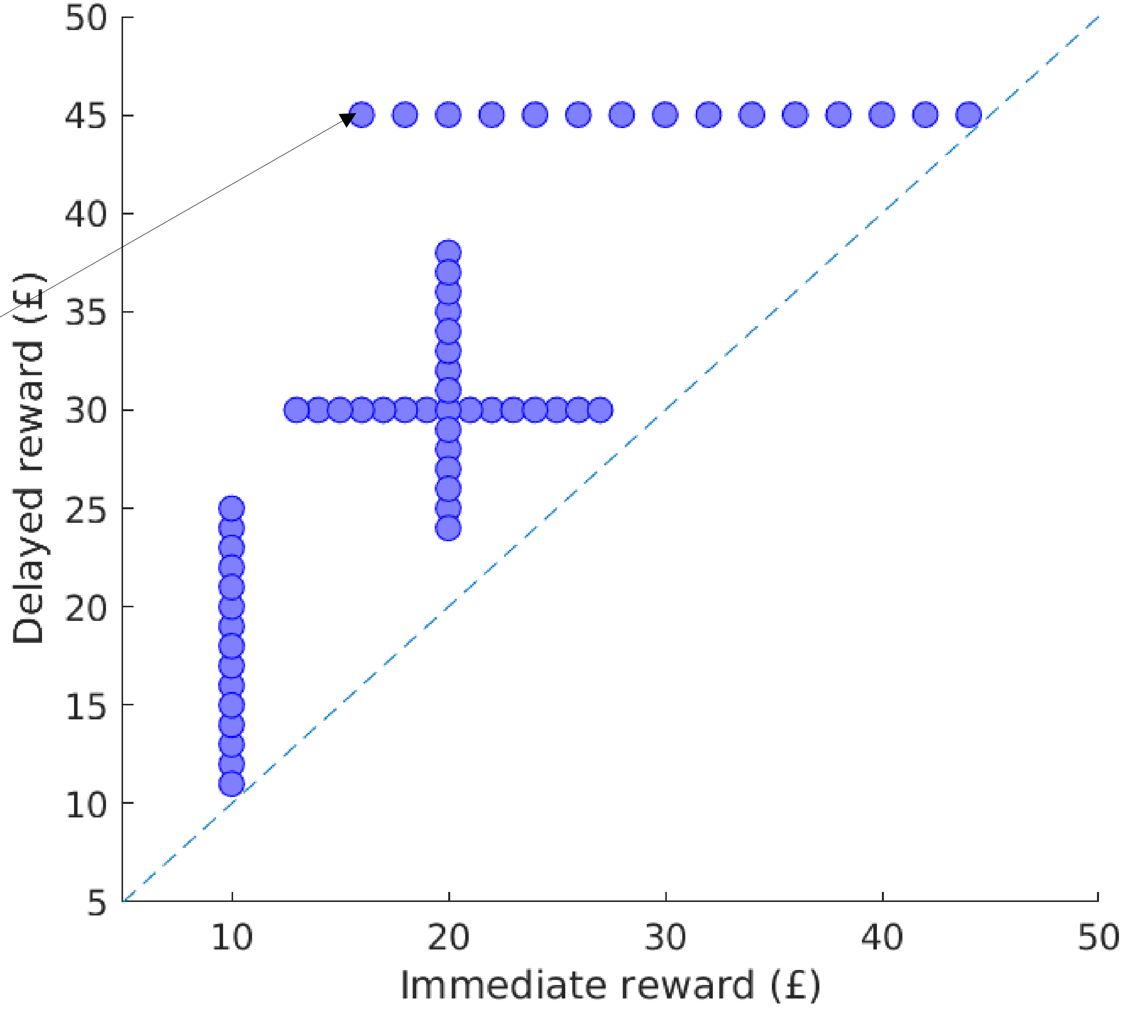
Example: temporal discounting task

- 60 choice pairs
  (each dots represents a pair of immediate & delayed rewards)
- 120 trials in total
  (each pairs repeated twice)



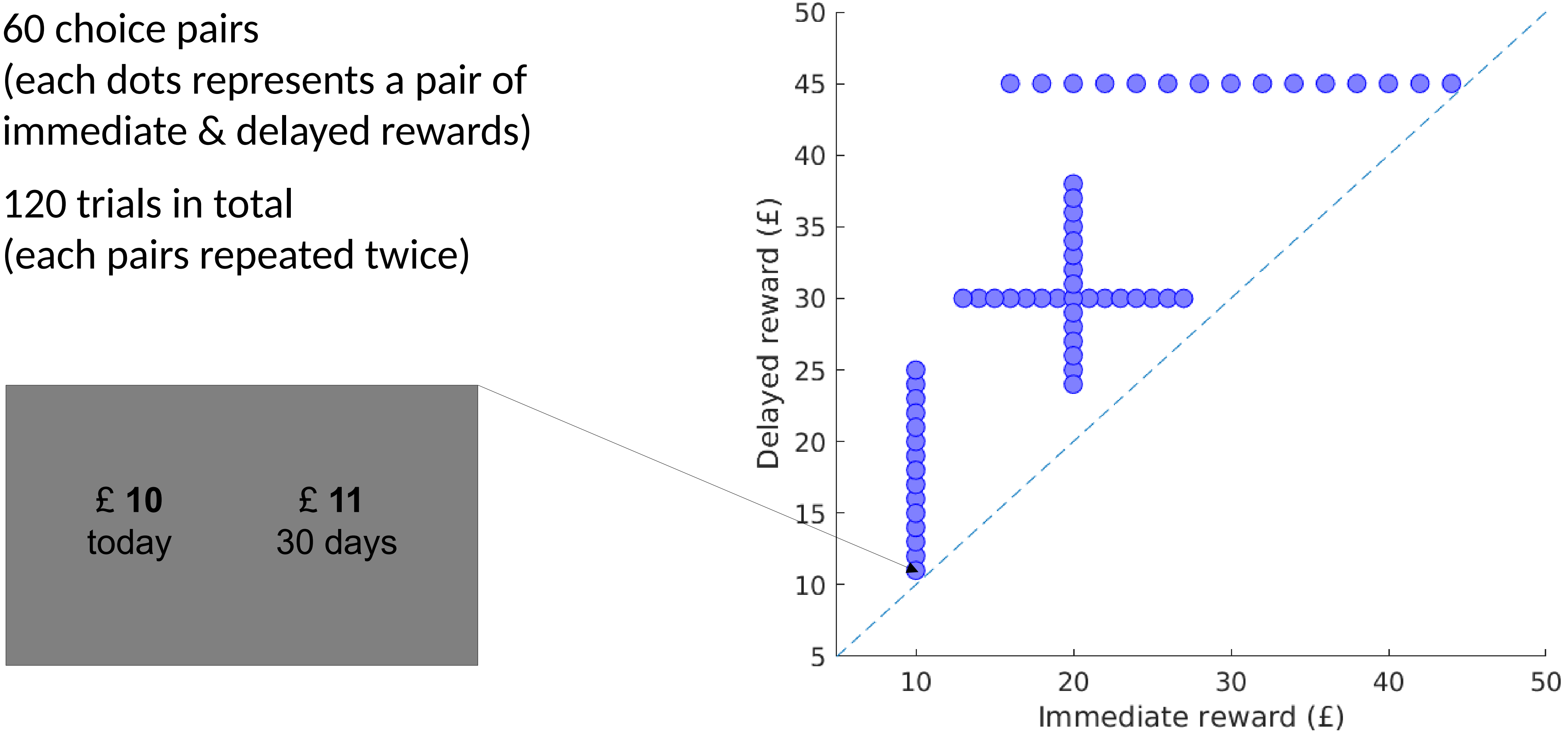
Example: temporal discounting task

- 60 choice pairs
  (each dots represents a pair of immediate & delayed rewards)
- 120 trials in total
  (each pairs repeated twice)



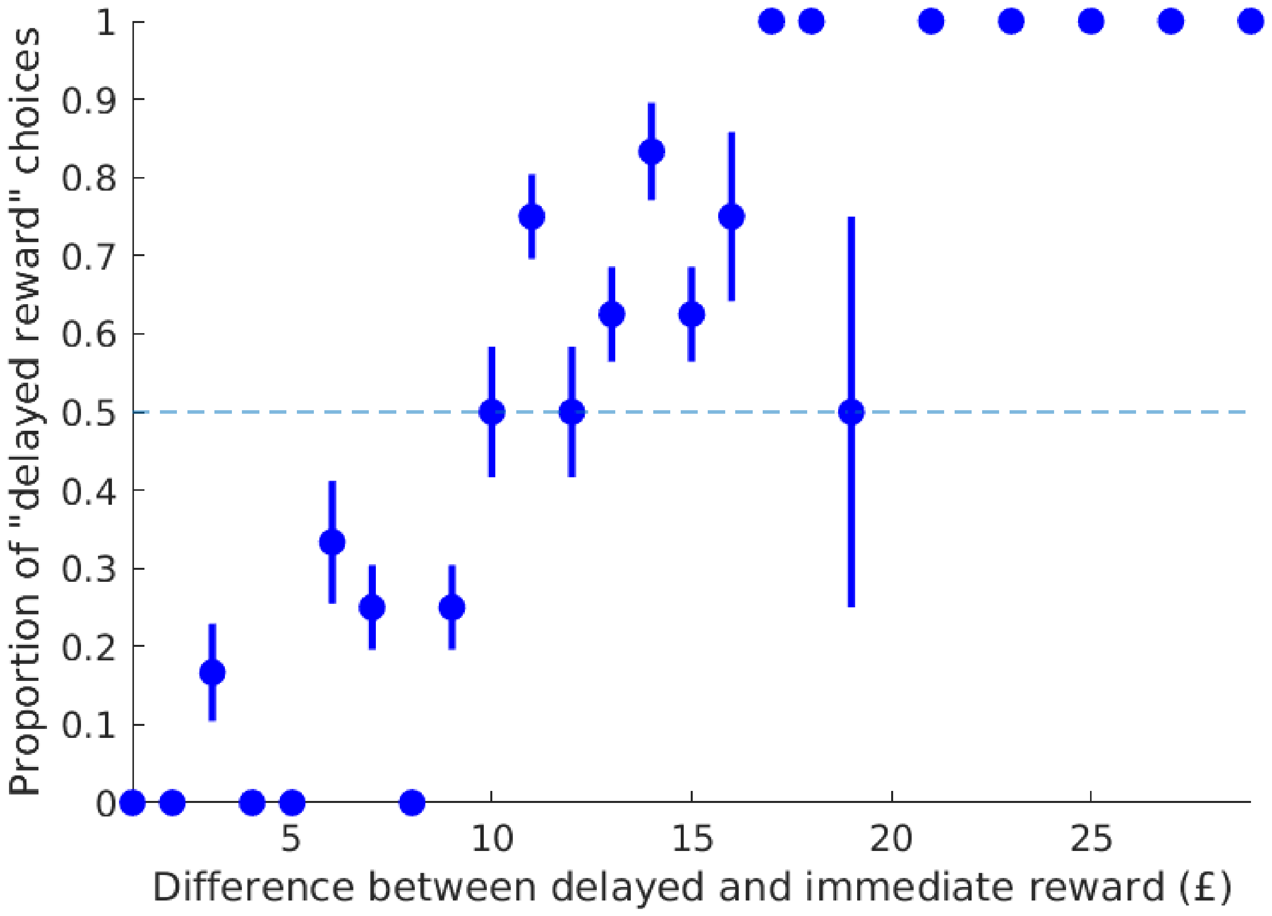
Example: temporal discounting task

- 60 choice pairs
  (each dots represents a pair of immediate & delayed rewards)
- 120 trials in total
  (each pairs repeated twice)



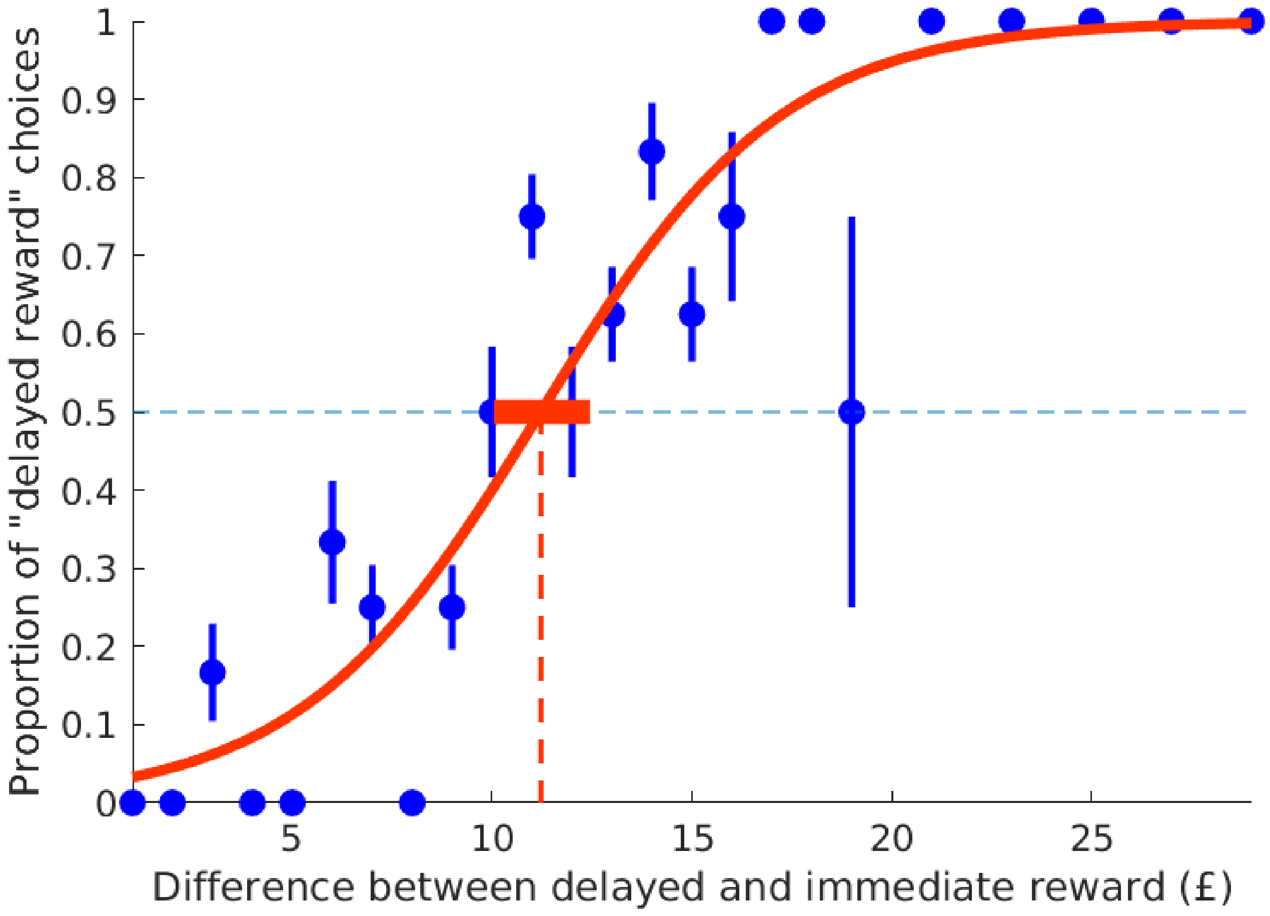
Example: temporal discounting task

- Example of data



Example: temporal discounting task

- Example of data



$$P(\text{choose delayed}) = \frac{1}{1 + e^{-\beta(x-\alpha)}}$$

x reward difference (delayed minus immediate)

α point of subjective equality of immediate and delayed reward

β slope of the function (control steepness)

Maximum likelihood estimation (MLE)

- Find the parameter values that maximize the probability (likelihood) of the data (given the model)
- The probability of the data is given by

$$P(\text{choose delayed}) = \frac{1}{1 + e^{-\beta(x-\alpha)}}$$

$$P(\text{choose immediate}) = 1 - P(\text{choose delayed})$$

- In practice, we code the binary choice immediate/delayed with a numerical variable (call it y) taking only 1 and 0 as values (e.g. representing choices of delayed and immediate rewards, respectively).

Maximum likelihood estimation (MLE)

- The probability of the data as a whole is obtained by multiplying the probability of individual responses

$$P(y_1, y_2, \ldots, y_n \mid \alpha, \beta) = P(y_1 \mid \alpha, \beta) P(y_2 \mid \alpha, \beta) \ldots P(y_n \mid \alpha, \beta)$$

- In practice, we usually take the logarithm of the likelihood, because it transform the product into a sum; we call this the *log-likelihood function*

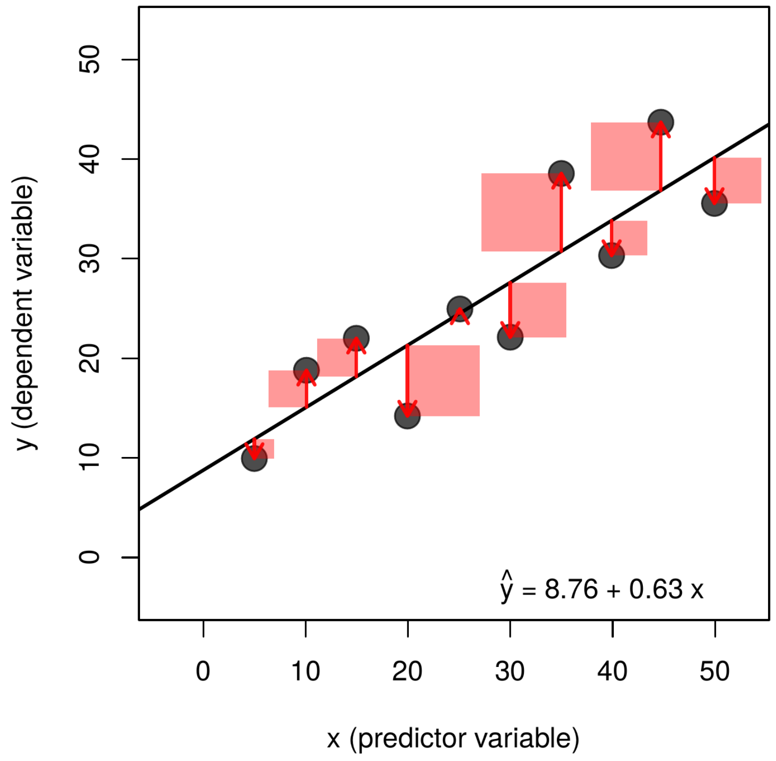
$$\mathcal{L}(y_1, y_2, \ldots, y_n \mid \alpha, \beta) = \log[P(y_1 \mid \alpha, \beta)] + \log[P(y_2 \mid \alpha, \beta)] + \ldots + \log[P(y_n \mid \alpha, \beta)]$$

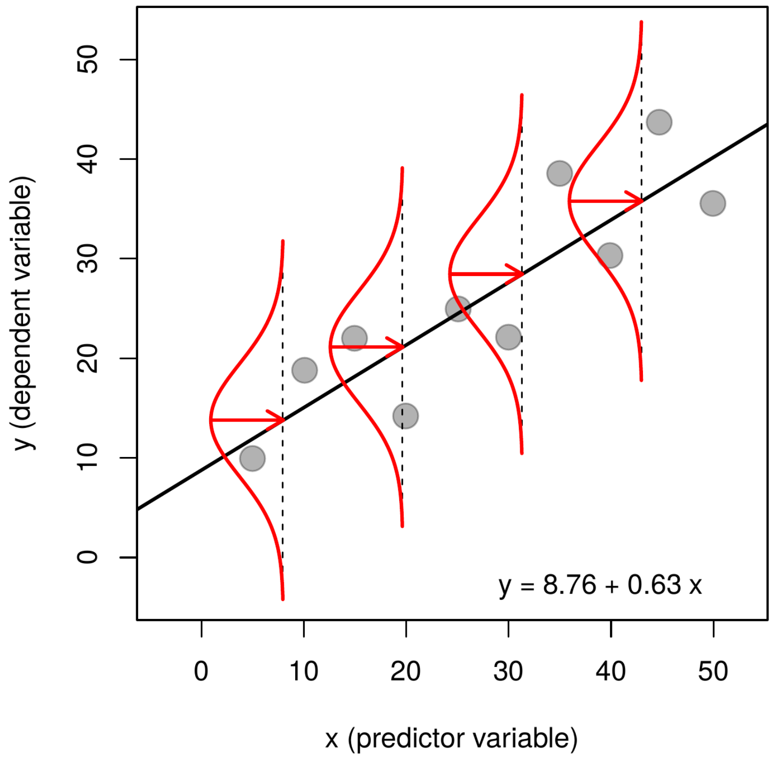
Maximum likelihood estimation (MLE)

- In words: *we find the values of the parameters maximize the log-likelihood function, which is the sum of the log-probabilities of individual trials/responses/datapoints*.
- For simple models, there may be formulas that give us the exact values that maximize the likelihood.
- In general, we often use *numerical optimization algorithms* that iteratively test different values of the parameters and find the maximum using a trial-and -error procedure (sort of); for example see function `fminsearch()` in Matlab, or function `optim()` in R
- The approach is general and can be used to fit *any* model
(as long as you can write and compute the likelihood function)

Maximum likelihood estimation (MLE)

- Least square estimation, used in linear regression, is a special case of MLE in which the probability of the data is given by a Gaussian distribution





Benefits of MLE estimation

MLE is popular because is a principled method to estimate parameters with some useful theoretical properties:

Consistency: as amount of data increases, MLE estimates become increasingly accurate, eventually converging to the true value.

Efficiency: MLE often makes the best use of available data, producing estimates with the lowest possible uncertainty.

Provides a natural way to compare models using tools like *likelihood ratio tests* or *AIC*.tes

MLE summary

- To fit *any* model on data:

1. Write a function that computes the probability of the data, given some values of the model parameters.
2. The parameters values that maximize such probability are the *maximum likelihood estimates* of the parameter.
3. Assess the uncertainty around such values (multiple ways of doing this – today we will use an approach known as *parametric bootstrapping*).

Possible data analysis report for portfolio

The dataset discounting_all.csv contains the data of 20 participants who completed this task:

- 10 healthy controls
- 10 Parkinson patients on dopaminergic medication (which is known to affect reward processing)

Research question to address:
Do medicated Parkinson patients differs from controls in their temporal discounting of economic value?