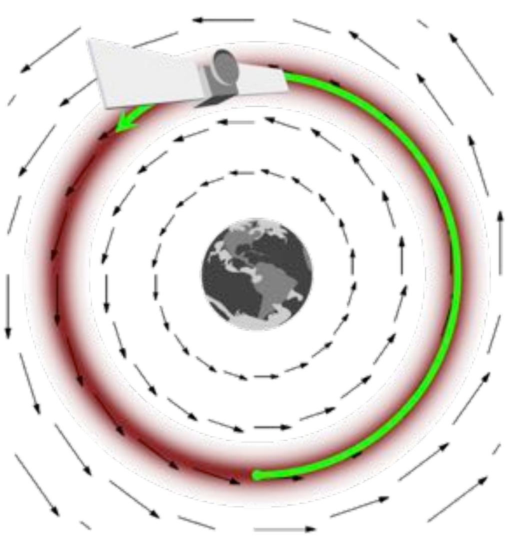
Stan Applications in Physics: Testing Quantum Mechanics and Modeling Neutrino Masses

Talia E. Weiss
Massachusetts Institute of Technology
StanCon, January 10, 2018

Stan and Physics:
A symbiotic relationship

Physics (especially statistical mechanics) helped fuel the development of MCMC methods.

Hamiltonian Monte Carlo employs concepts from physics: Energy, position, momentum …



Why use physics?

Probabilistic systems

↕

Physical systems in equilibrium

Image: Betancourt (2017)

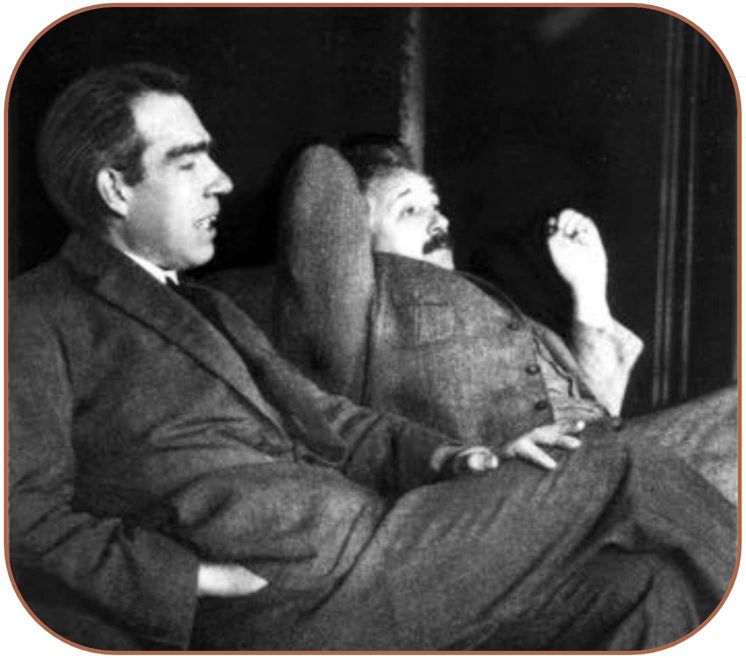
Stan and Physics:
A symbiotic relationship

Part 1: Testing interpretations of quantum mechanics

Does quantum weirdness exist?

Part 2: Modeling the neutrino mass problem

Can we experimentally determine the neutrino masses?



We employed Stan:

– To generate pseudo-data
– To extract physical parameters from data

Image: Wikipedia

Applying Stan to advance experimental physics: A few observations

Stan is a powerful tool for physicists because it enables:

- Modeling complex systems

  While accounting for many sources of uncertainty

  Can capitalize on detailed experimental knowledge

- Extracting physical parameters from data

  While incorporating known information as priors

  Precludes need to create own "fitters"

Applying Stan to advance experimental physics: A few observations

Stan is a powerful tool for physicists because it enables:

- Modeling complex systems
  While accounting for many sources of uncertainty
  Can capitalize on detailed experimental knowledge
- Extracting physical parameters from data
  While incorporating known information as priors
  Precludes need to create own "fitters"

Challenges: Modeling many systematics (& computing time); difficult normalizations

Morpho: Our python-based wrapper for Stan

Morpho bridges between Stan/PyStan
and data input/output.

Features include:

- Loading data & Stan functions
- Loading & outputting to root, hdf5 and R files
- Running Stan diagnostic tests
- Creating plots (histograms, divergence plots, etc.)

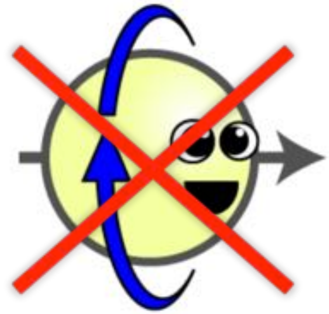
Stan application #1:
A test of quantum mechanics with neutrinos

J. A. Formaggio, D. I. Kaiser, M. M. Murskyj, and T. E. Weiss, "Violation of the Leggett-Garg inequality in neutrino oscillations," *Physical Review Letters*, vol. 117, no. 5, Article ID 050402, 2016.

Collaborators

Quantum mechanics and the elements of "quantum weirdness"

In the **quantum realm** of particles:



- No *definite* properties, like "moving fast" or "spinning clockwise"
- Instead, quantum mechanics yields only probabilities: $P(\text{clockwise}, t)$

Images: Quantum Diaries, Wired

Quantum mechanics and the elements of "quantum weirdness"

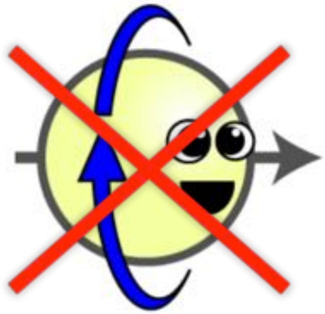
In the **quantum realm** of particles:

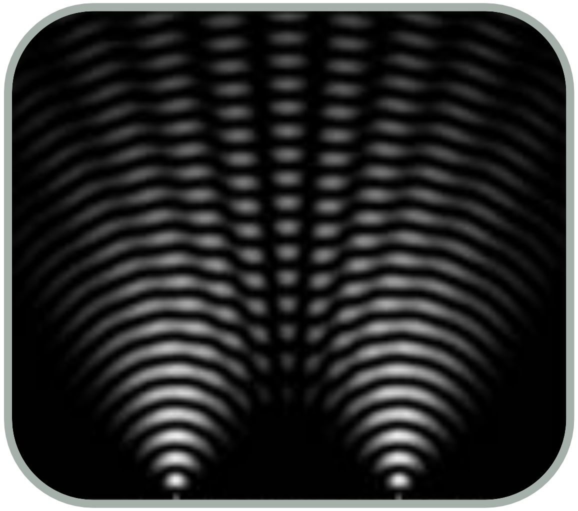
– No *definite* properties, like "moving fast" or "spinning clockwise"

– Instead, quantum mechanics yields only probabilities: $P(\text{clockwise}, t)$

– Systems exist in superpositions: possess "incompatible" properties

– They sometime act "wave-like"; other times, "particle-like"





Interference

Images: Quantum Diaries, Wired

Is quantum weirdness real?
(Schrodinger and Einstein: “No way.”)

Copenhagen (*et al.*):
Quantum weirdness is very real.

Schrodinger: Superposition is as absurd as a cat that’s both dead and alive.

Einstein (*et al.*):
The universe is deterministic → systems have hidden variables.

Is quantum weirdness real?
(Schrodinger and Einstein: “No way.”)

Copenhagen (*et al.*):
Quantum weirdness is very real.

Schrodinger: Superposition is as absurd as a cat that’s both dead and alive.

Einstein (*et al.*):
The universe is deterministic → systems have hidden variables.

Who was right?

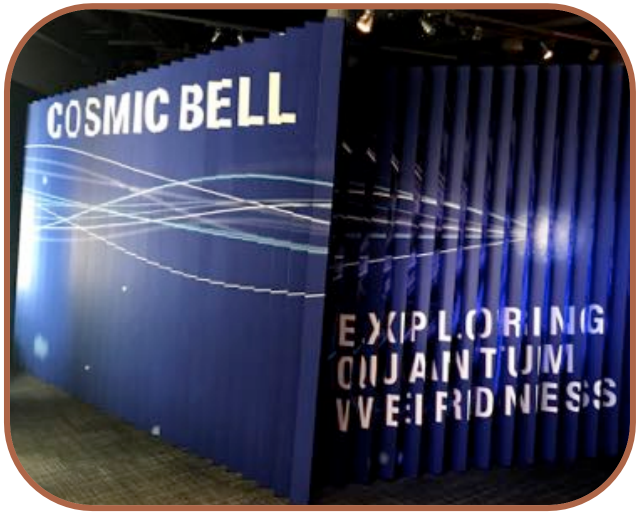
Using Stan, we can model systems with and without <u>superposition</u>—and compare both to data.

Testing quantum interpretations: A search for strange correlations

Superposition → Correlated properties in space (entanglement) and time

Testing quantum interpretations: A search for strange correlations

Superposition → Correlated properties in space (entanglement) and time

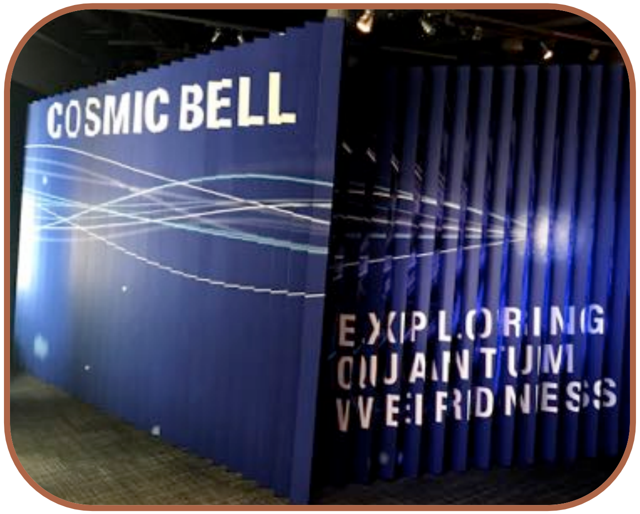


Space: Bell's inequality (1964)

- Violation → "spooky action"
- All tests *thus far* found violations

Testing quantum interpretations: A search for strange correlations

Superposition → Correlated properties in space (entanglement) and time



Space: Bell's inequality (1964)

- Violation → "spooky action"
- All tests *thus far* found violations

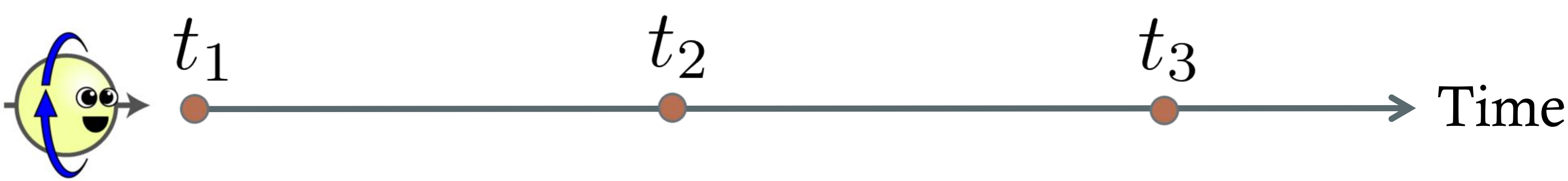
Time: $\boxed{\text{Leggett-Garg inequality (1985)}}$

- Violation → superposition + time evolution

Probing quantum time evolution with Leggett-Garg inequalities (LGIs)

Two-state observable $\hat{Q}$:

$$\hat{Q}|\text{spin-}\!\uparrow\rangle = +1|\text{spin-}\!\uparrow\rangle$$
$$\hat{Q}|\text{spin-}\!\downarrow\rangle = -1|\text{spin-}\!\downarrow\rangle$$



$$\hat{Q}(t_1) \longrightarrow \hat{Q}(t_2) \longrightarrow \hat{Q}(t_3)$$

Constructing the Leggett-Garg parameter from measurements

Two-measurement **correlation** (joint probability):

$$C_{ij}\left(\hat{Q}(t_i), \hat{Q}(t_j)\right)$$

Constructing the Leggett-Garg parameter from measurements

Two-measurement **correlation** (joint probability):

$$C_{ij}\left(\hat{Q}(t_i), \hat{Q}(t_j)\right)$$

LG parameter for **n** measurements:

$$\boxed{K_n \equiv \sum_{i=1}^{n-1} C_{i,i+1} - C_{n,1}}$$

Constructing the Leggett-Garg parameter from measurements

Two-measurement **correlation** (joint probability):

$$C_{ij}\left(\hat{Q}(t_i), \hat{Q}(t_j)\right)$$

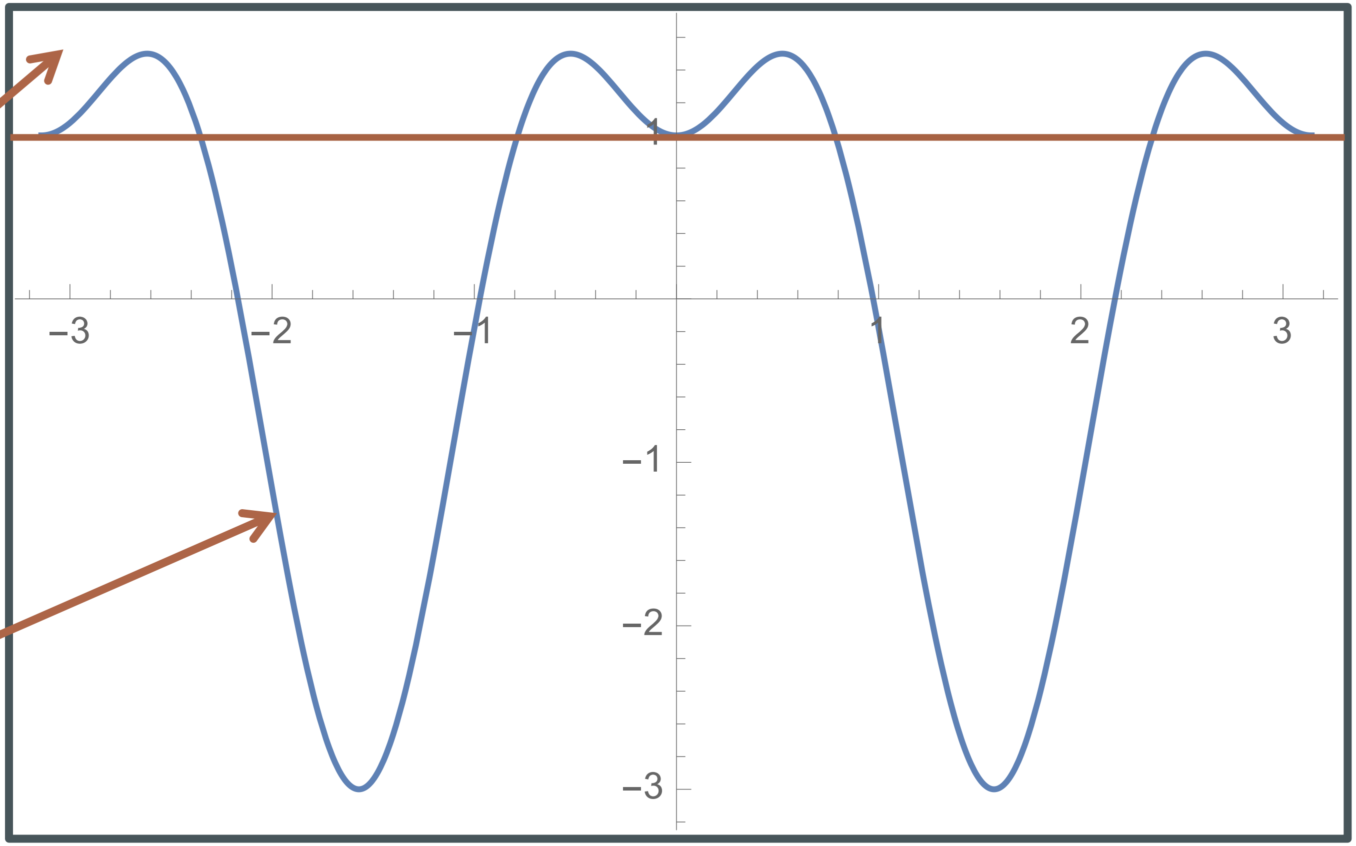
LG parameter for **n** measurements:

$$\boxed{K_n \equiv \sum_{i=1}^{n-1} C_{i,i+1} - C_{n,1}} \xrightarrow[\text{(“classical”)}]{\textit{No superposition}}$$

$$K_n^C = \sum_{i=1}^{n-1} C_{i,i+1} - \prod_{i=1}^{n-1} C_{i,i+1} \boxed{\leq n-2}$$

If "quantum weirdness" exists, quantum systems violate the LGIs

LGI violating region



K_3^Q (LG parameter for a quantum system)

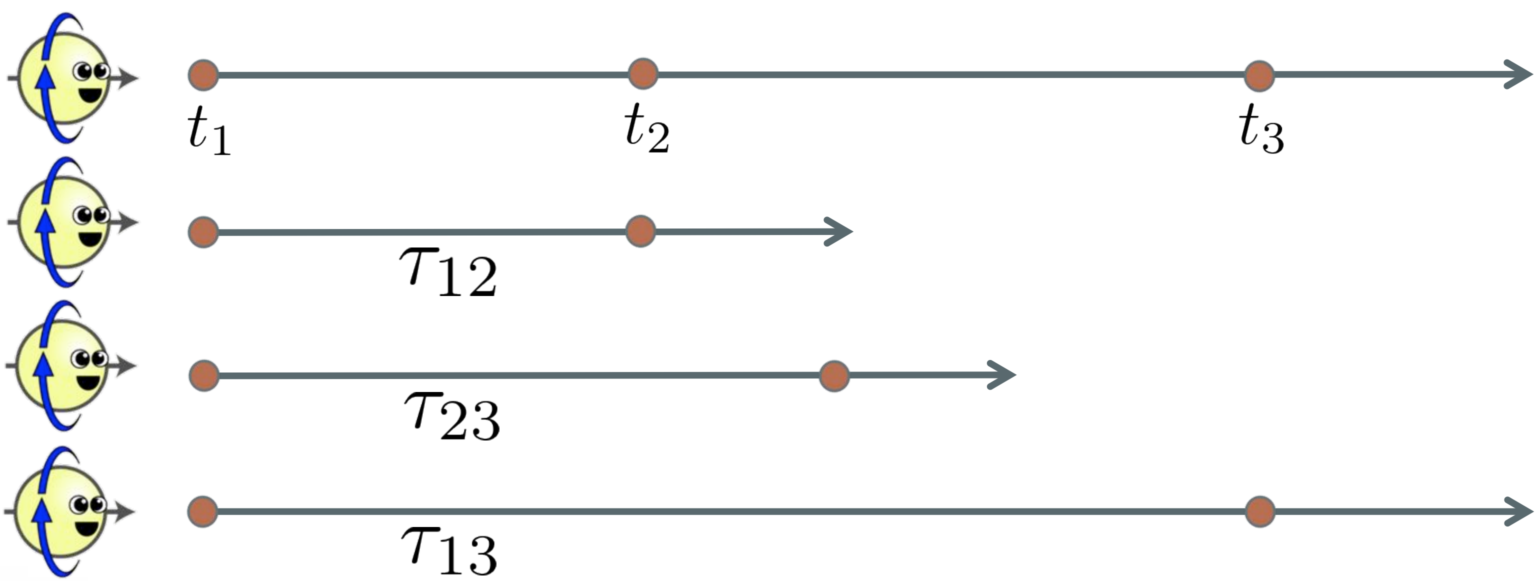
Time between measurements

How do we stop invasive measurements from creating (fake) LGI violations?

Stationarity: correlations C_{ij} depend only on $\tau = t_j - t_i$

AND

Prepared ensemble: Perform measurements on identically prepared systems



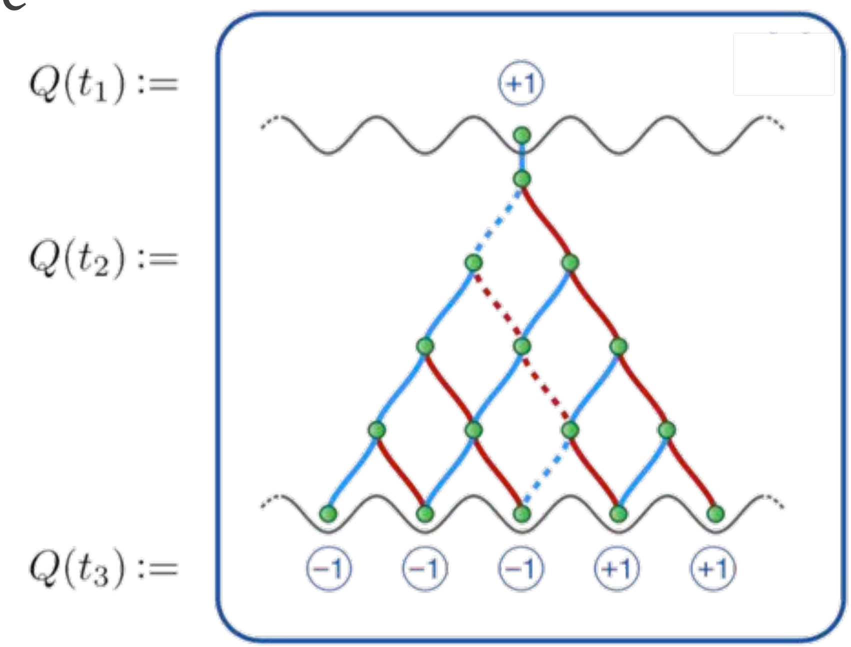
Huelga et al. (1996)

Previous LGI research:
many violations, some loopholes, & no neutrinos

LGI violation observed with:

- Nuclear Magnetic Resonance
- Superconducting qubits
- P impurities in Si

No one had yet tested LGIs with neutrinos (… or Stan).



Mapping a Cs atom's quantum walk to an LGI formalism (*Robens et al., 2014*)

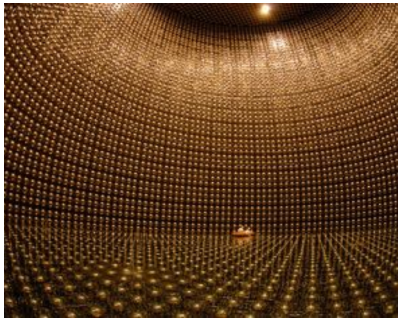
The world is awash with neutrinos: ghost-like, shape-shifting particles

"In scarcely more than half a century, neutrinos have gone from wispy, exotic particles at the edge of detectability to tools for investigating matter at its most essential." -- David Kaiser, MIT

The world is awash with neutrinos: ghost-like, shape-shifting particles

"In scarcely more than half a century, neutrinos have gone from wispy, exotic particles at the edge of detectability to tools for investigating matter at its most essential." -- David Kaiser, MIT

1956: Electron neutrino detected



2015: Nobel Prize awarded for neutrino oscillations

1933: Fermi proposed the neutrino—a (missing) neutral particle

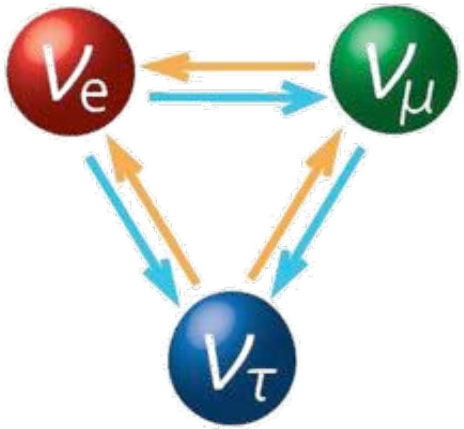
1998-2001: Super-K, SNO discovered that neutrinos have mass (oscillations)

Quote: Aeon, "Operation: Neutrino"

Images: Wikipedia, Florida Tech

Neutrino oscillations: a case of quantum weirdness

Neutrinos come in 3 **flavors**: $\nu_e, \; \nu_\mu, \; \nu_\tau$



Each flavor state is a superposition of 3 **mass states:** $m_1, \; m_2, \; m_3$

"Two-Neutrino Model"

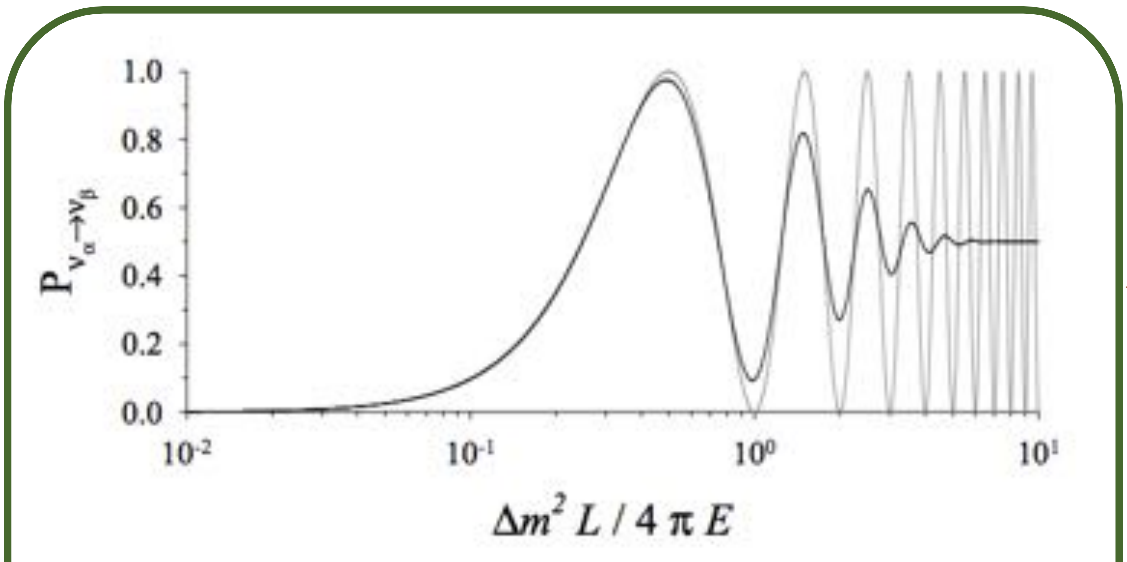
Oscillations between flavors occur with probability:

$$P_{e\mu} \equiv P_{\nu_e \rightarrow \nu_\mu} \propto \sin^2\left(\frac{\Delta m_{12}^2 L}{4E}\right)$$

Image: BBC

Goal: Use *neutrino oscillation data* to test for LGI violations

Investigate validity of (a class of) *hidden variable theories*



Two-neutrino oscillation probability

Constructing a neutrino-based Leggett-Garg parameter

Goal: Use **neutrino oscillation data** to test for LGI violations.

Flavor operator (observable) $\hat{Q}(t)$:

$$\hat{Q}\,|\nu_\mu\rangle = +1\,|\nu_\mu\rangle$$
$$\hat{Q}\,|\nu_e\rangle = -1\,|\nu_e\rangle$$

For a beam of ν_μ in vacuum, we found **correlations** between flavor measurements in in terms of ν_μ "survival" probabilities:

$$\boxed{C_{ij} = 2P_{\mu\mu} - 1}$$

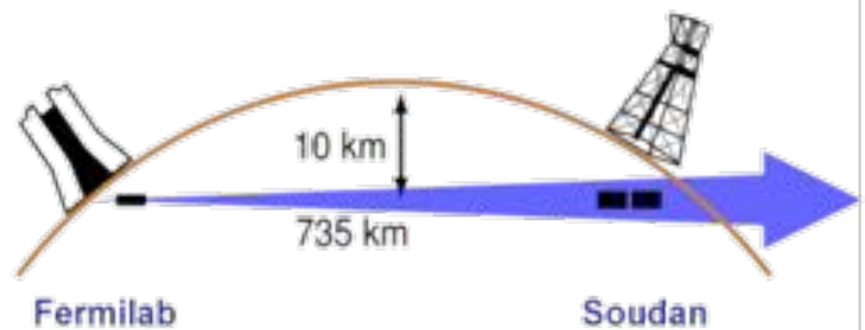
A crucial change in formalism:
neutrino energy replaces *time between measurements*

Correlations depend only on phases —not on individual times

$$\psi_{a;ij} \equiv \frac{\Delta m^2 \tau_{ij}}{4\hbar E_a}$$

For neutrinos propagating at light speed across a fixed baseline,

$$\tau_{ij} = \delta L/c.$$



So $C_{ij} = C(E_a)$ → consider measurements taken with respect to neutrino energy (not time)

Image: APS Physics

Neutrino-based LG parameter: the basis of a *quantum model*

For a *beam of neutrinos* travelling a fixed distance,
we can calculate a **Leggett-Garg parameter**:

$$K_n^Q(C) = K_n^Q(P_{\mu\mu}) = K_n^Q(\psi_a)$$

Neutrino-based LG parameter: the basis of a *quantum model*

For a *beam of neutrinos* travelling a fixed distance, we can calculate a **Leggett-Garg parameter**:

$$K_n^Q(C) = \boxed{K_n^Q(P_{\mu\mu})} = K_n^Q(\psi_a)$$

We use this formula to:

A. **Generate pseudo-data in Stan** for neutrinos exhibiting superposition

Neutrino-based LG parameter: the basis of a *quantum model*

For a *beam of neutrinos* travelling a fixed distance, we can calculate a **Leggett-Garg parameter**:

$$K_n^Q(C) = \boxed{K_n^Q(P_{\mu\mu})} = K_n^Q(\psi_a)$$

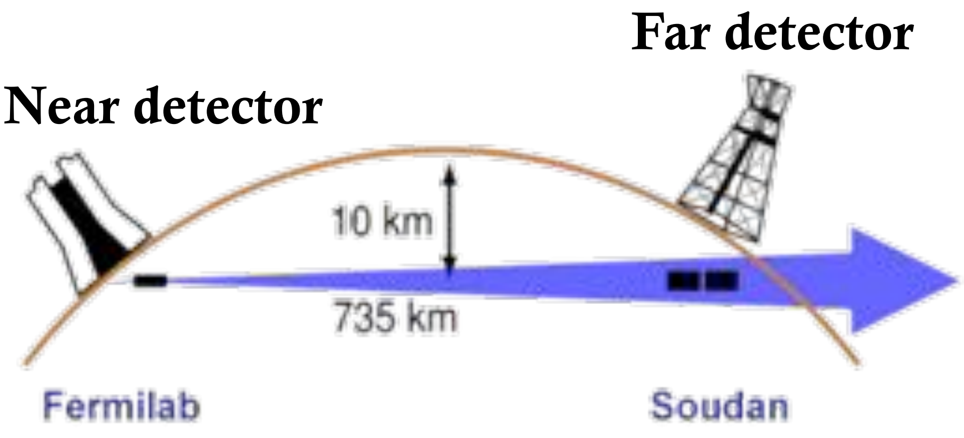
We use this formula to:

A. **Generate pseudo-data in Stan** for neutrinos exhibiting superposition

B. **Compute *K*** from oscillation data

➡ We expect LGI violations for particular ψ_a

Data from the MINOS Experiment:
Longest length scale for a test of quantum weirdness



98% $\boldsymbol{\nu_\mu}$ **Neutrino beam** → **Some** $\boldsymbol{\nu_\mu}$**,** **Some** $\boldsymbol{\nu_e}$

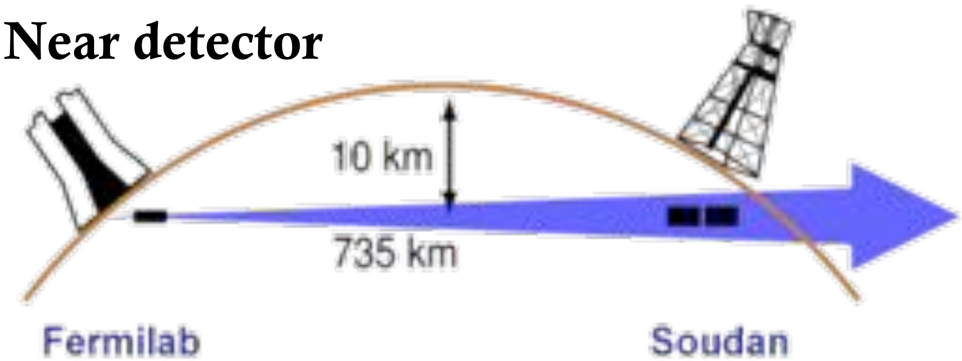
Survival probabilities ($P_{\mu\mu}$) measured in Soudan, MN

Images: APS Physics, Fermilab

Data from the MINOS Experiment:
Longest length scale for a test of quantum weirdness

Far detector

Near detector



Survival probabilities ($P_{\mu\mu}$) measured in Soudan, MN

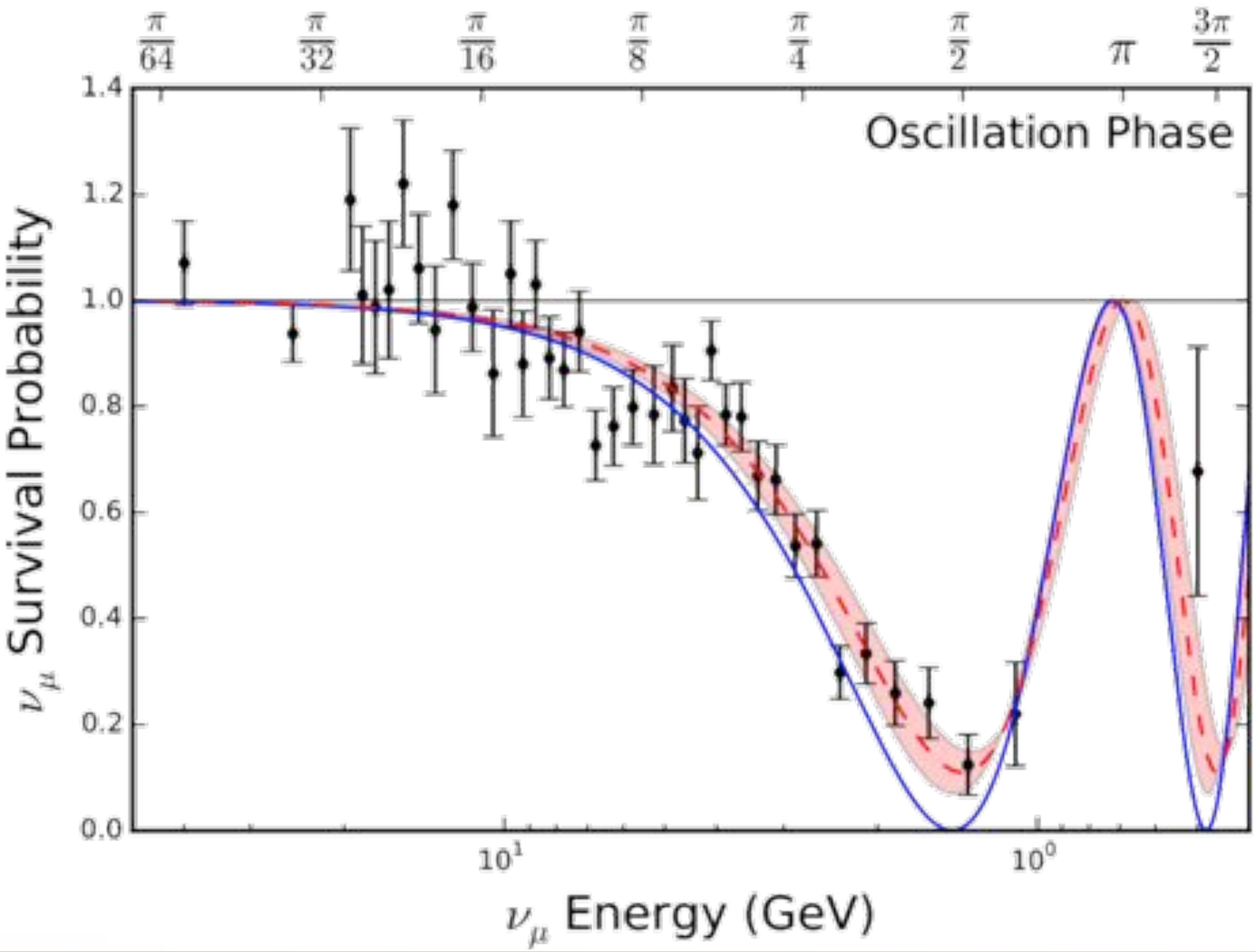
98% ν_μ **Neutrino beam** **Some ν_μ, Some ν_e**

1. <u>Prepared ensemble</u> assumption satisfied

2. <u>Stationarity</u> demonstrated for MINOS data (Adamson *et al.*, 2012)

Images: APS Physics, Fermilab

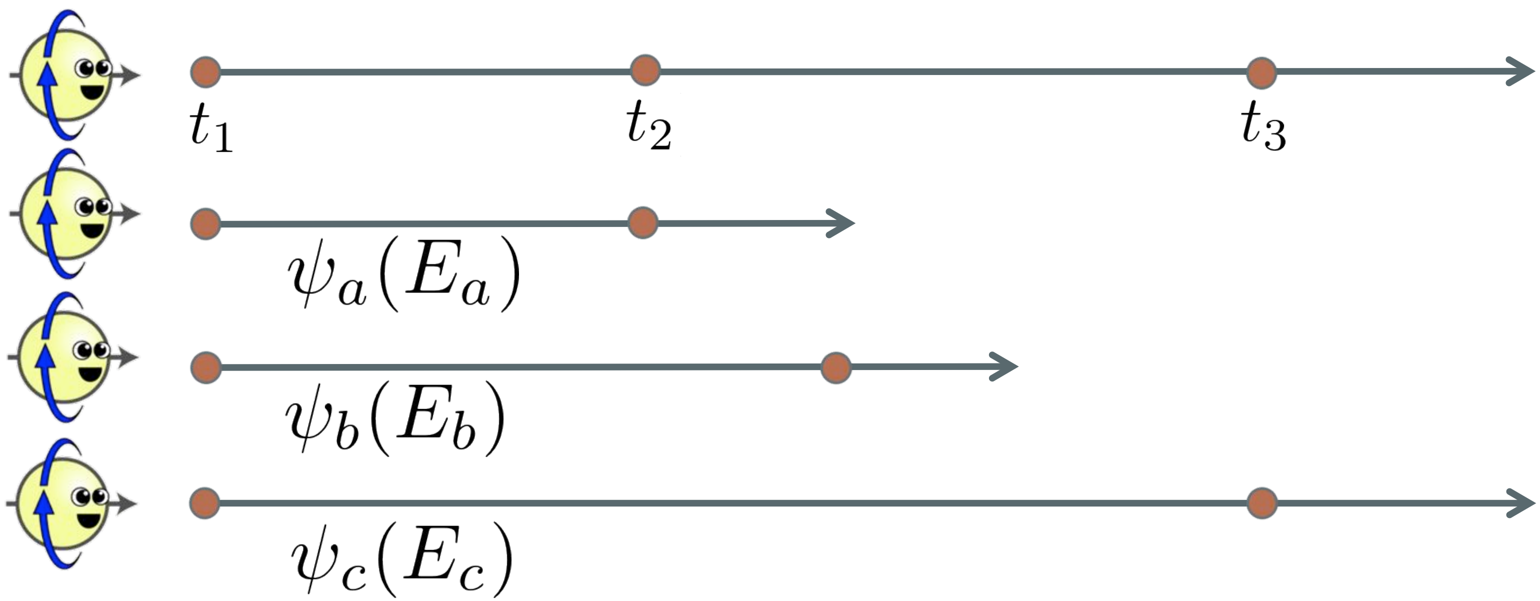
Neutrino oscillation data ($P_{\mu\mu}$) from the MINOS Experiment



Solid blue curve: Prediction for oscillations assuming global values of mixing parameters.

Dashed red curve: Prediction using mixing parameters **determined from MINOS data in Stan** with a $\sim 1\sigma$ confidence interval (red band).

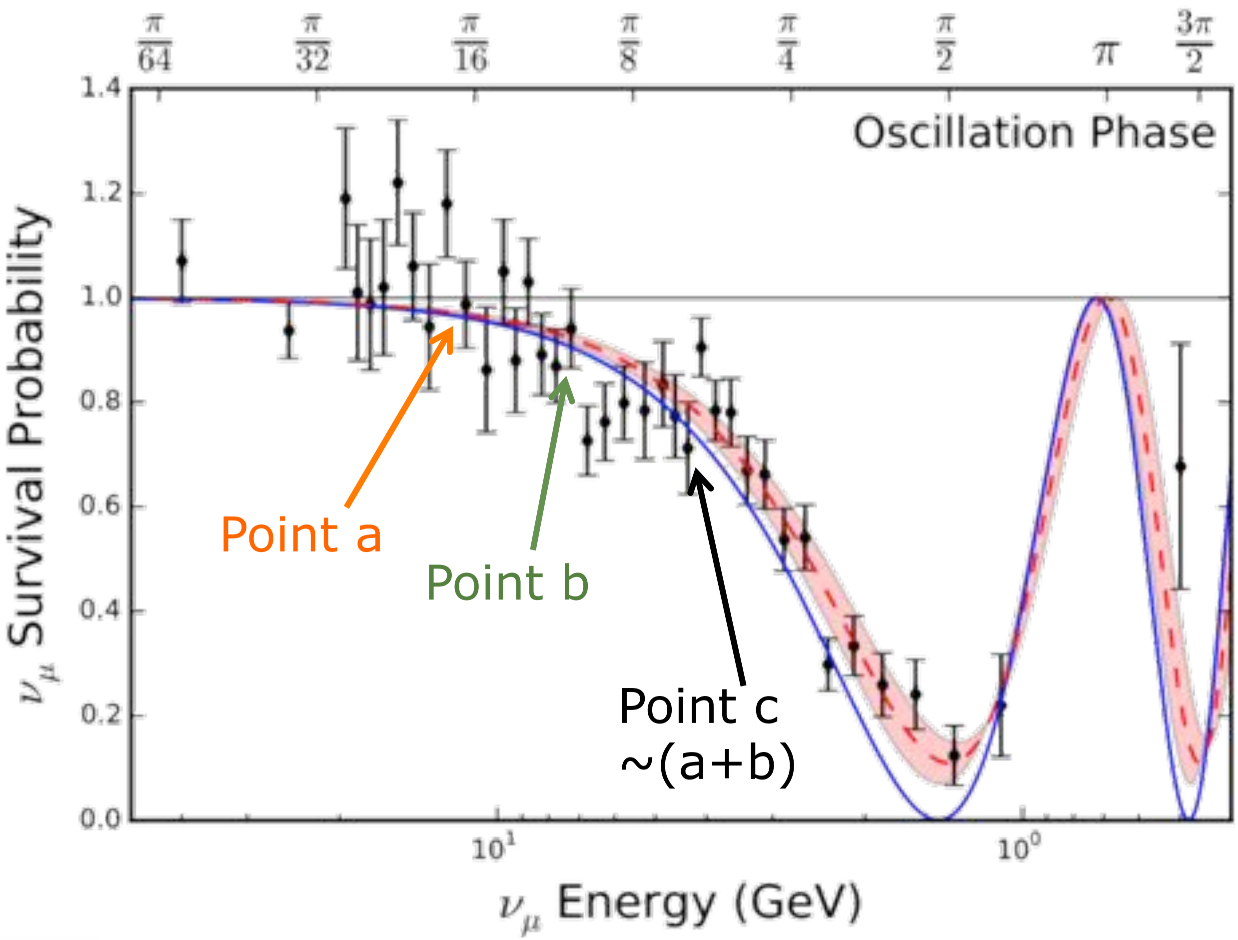
Computing "observed" Leggett-Garg parameters using Stan



Our data selection condition (for K_3):

$$\psi_a + \psi_b = \psi_c$$

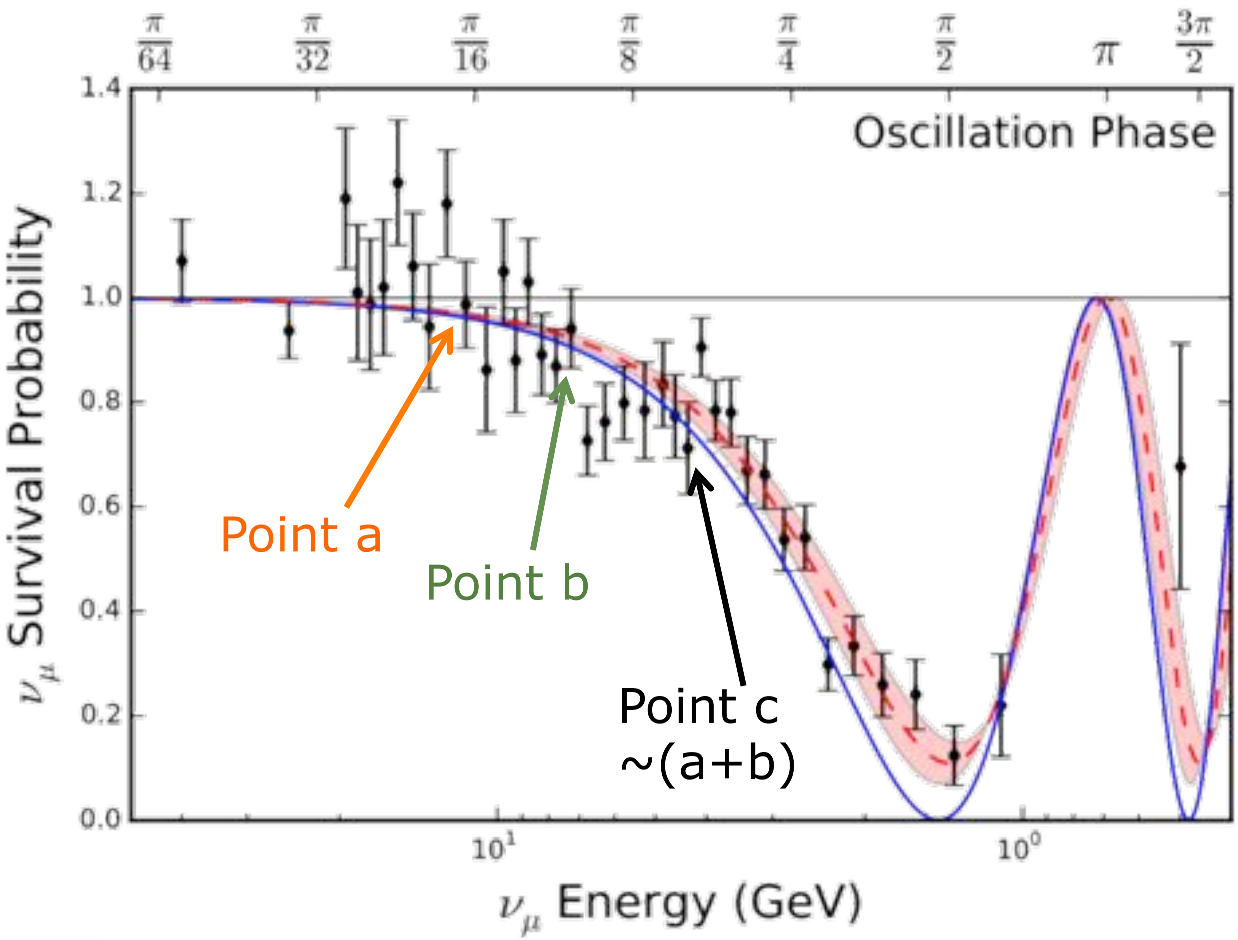
Computing "measured" Leggett-Garg parameters using Stan



$$\psi_a + \psi_b \simeq \psi_c$$

(within 0.5%)

Computing "measured" Leggett-Garg parameters using Stan



$$\psi_a + \psi_b \simeq \psi_c$$

(within 0.5%)

Using a function we wrote in Stan, we found 82 sets of points that met this condition for K_3, and 715 for K_4.

Accounting for false-positive LGI violations

	K_3	K_4
Total number of points	82	715
Number of LGI violations	64 (78.0%)	577 (80.1%)

Even if neutrinos were "classical" (no superposition), we would expect *some* LGI violations due to fluctuations above the limit.

How do we address these false-positives?

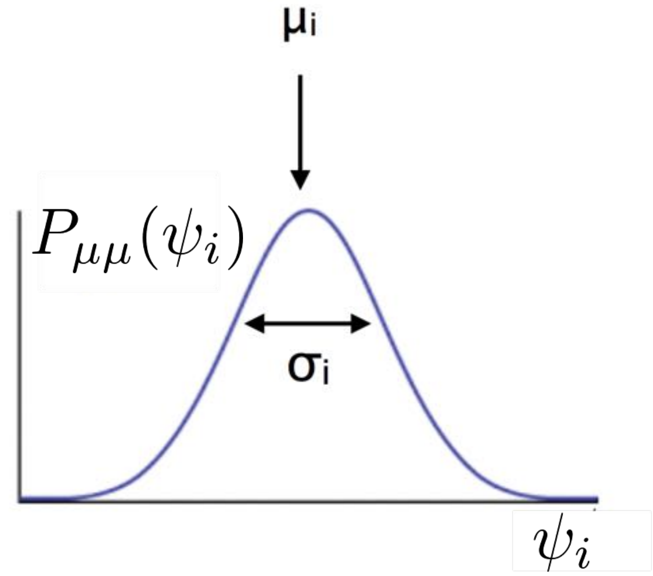
Generate **pseudo neutrino data**
Create **classical and quantum models** of K_3 and K_4
Predict **number of LGI violations** according to each model

Using Stan to model the expected number of violations—with and without superposition

1. Create large sample of **pseudo-data**:

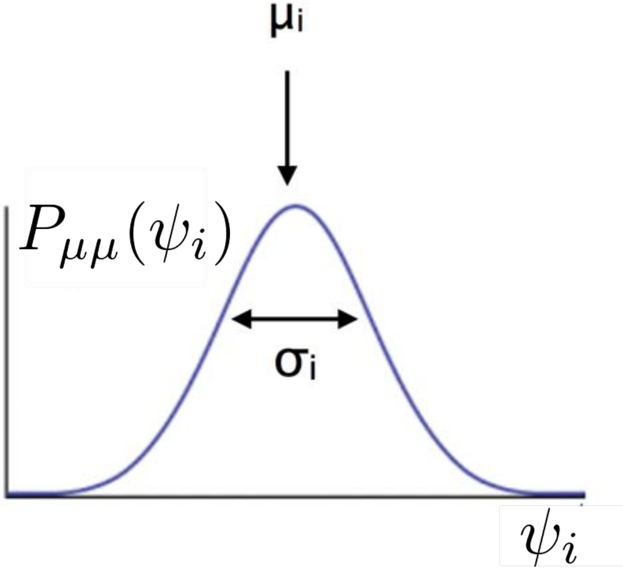
   Drawn from normal distributions with means, variances of observed $P_{\mu\mu}$

   → fake "experiments" with same correlations as data



Using Stan to model the expected number of violations—with and without superposition

1. Create large sample of **pseudo-data**:
   Drawn from normal distributions with means, variances of observed $P_{\mu\mu}$
   → fake "experiments" with same correlations as data



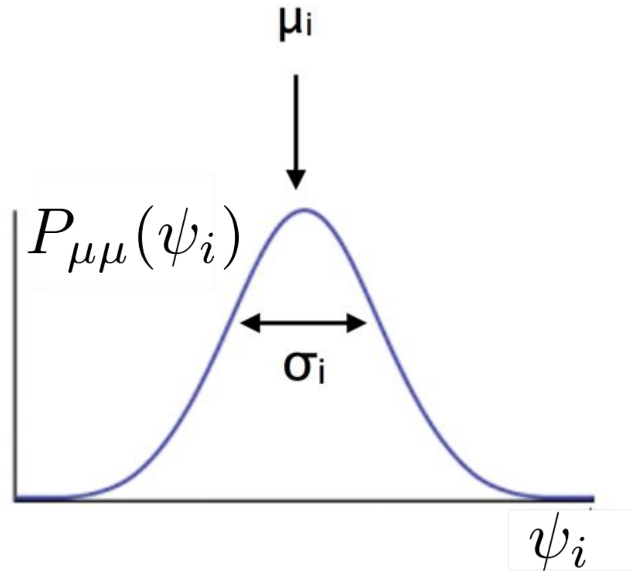
2. Calculate $K^C(P_{\mu\mu})$ and $K^Q(P_{\mu\mu})$ from pseudo-data

$$K_n^C = \sum_{i=1}^{n-1} \mathcal{C}_{i,i+1} - \prod_{i=1}^{n-1} \mathcal{C}_{i,i+1}$$

$$K_n^Q = \sum_{i=1}^{n-1} \mathcal{C}_{(i,i+1)} - \mathcal{C}_{(n,1)}$$

Using Stan to model the expected number of violations—with and without superposition

1. Create large sample of **pseudo-data**:
   Drawn from normal distributions with means, variances of observed $P_{\mu\mu}$
   → fake "experiments" with same correlations as data



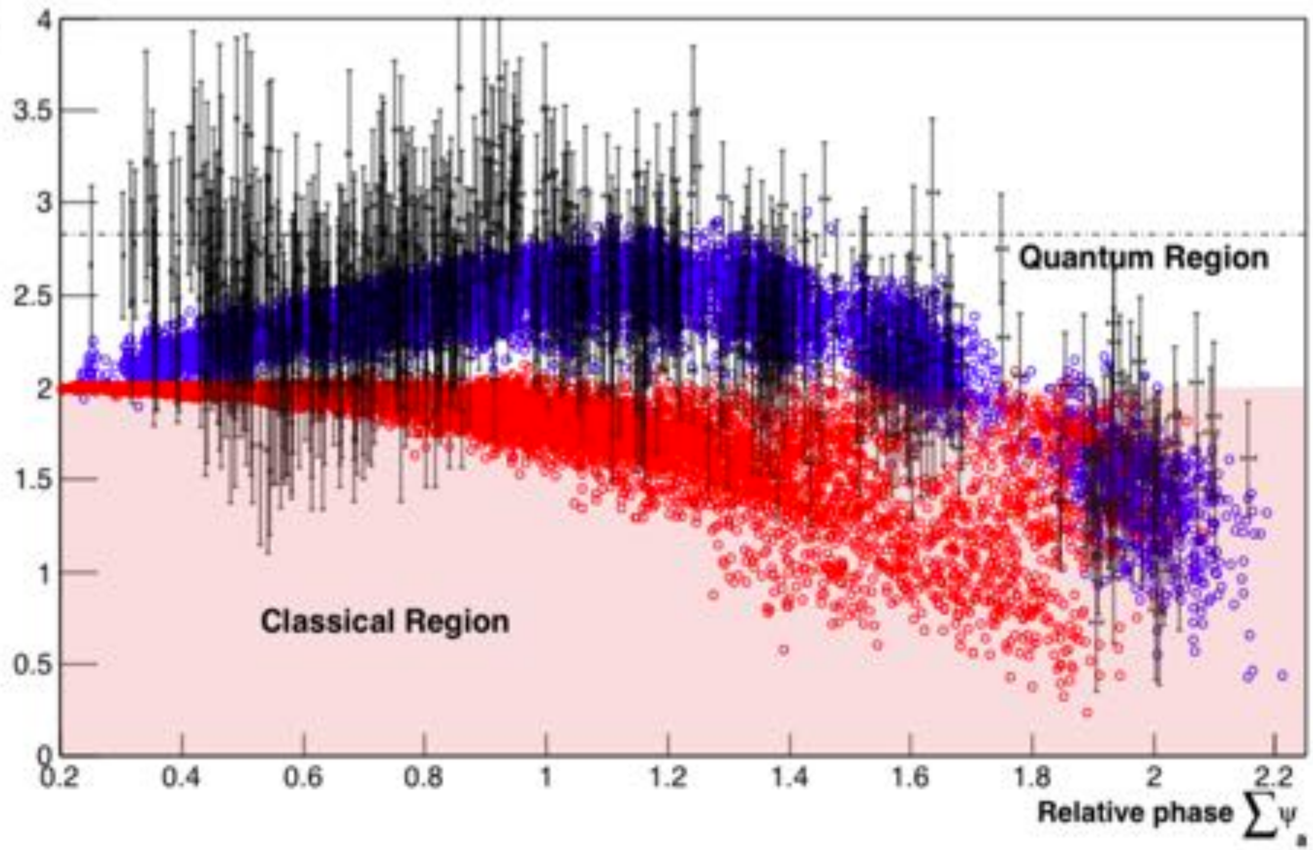
2. Calculate $K^C(P_{\mu\mu})$ and $K^Q(P_{\mu\mu})$ from pseudo-data

3. Extract parameters `n_violations_classical` and `n_violations_quantum`
   (Number of values in K^C and K^Q vectors that exceed the classical bound.)

K_4 values computed from MINOS data (as a function of the sum over phases)

Observed **K_4 values** (black points).

Superimposed on points simulated with a classical LGI model (red) or a quantum LGI model (blue).



The analysis: relevant Stan code excerpts

```
functions{
    real get_K_quantum(vector Pmumu, int[] phase_index){…};
    real get_K_class(vector Pmumu, int[] phase_index){…};
    ⋮
    real find_closest_point(real phase_sum, vector points,
        real tolerance){…};
}

data{
    int<lower=3,upper=4> nOrder; //Number of measurements
    int isQuantum; //Decide if reality is classical (hidden
                     variables) or quantum (superposition)
    ⋮
    //Data from MINOS: survival probabilities, phases, errors
    vector[nData] Prob;
    ⋮
}
```

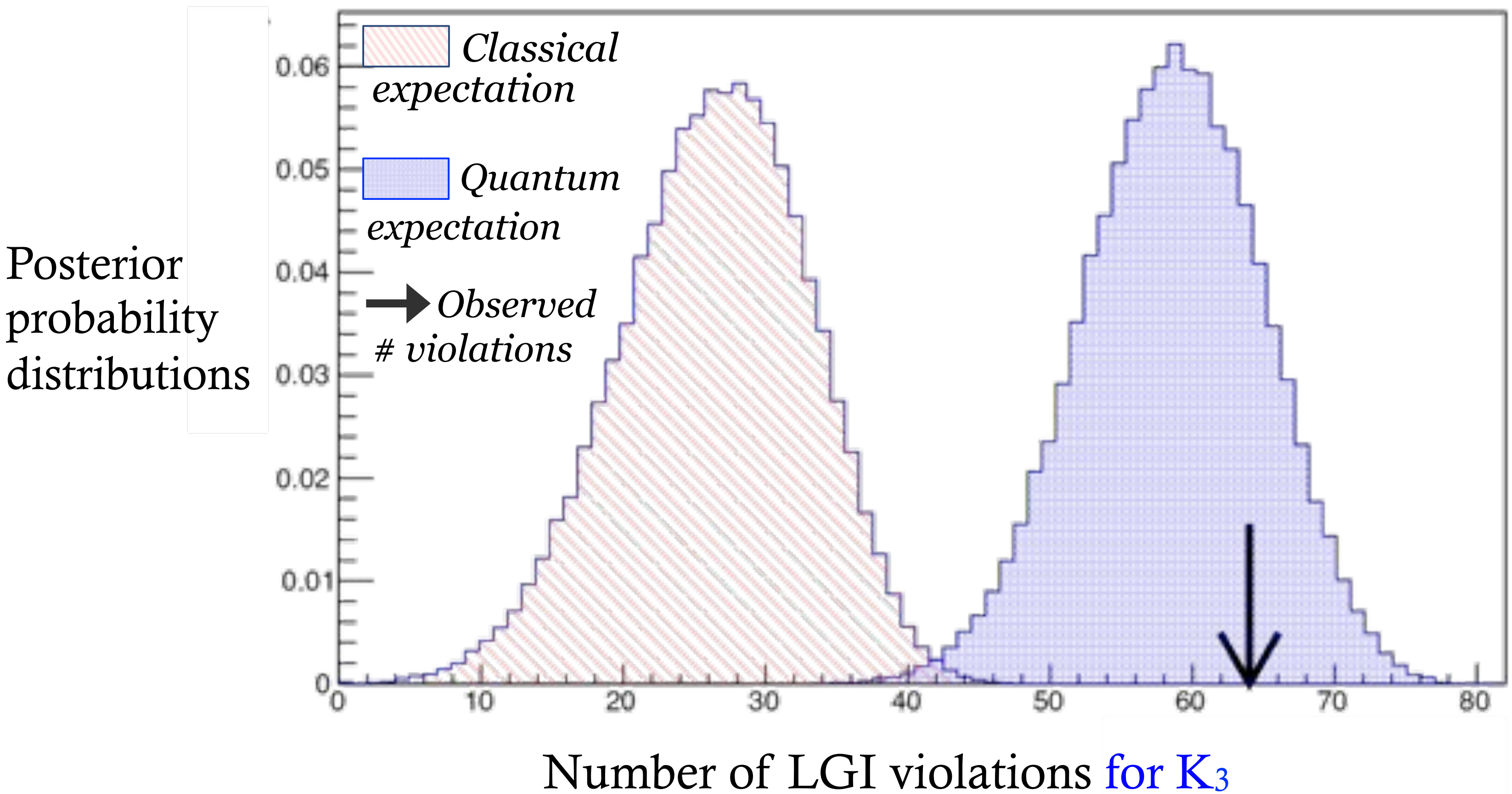
The analysis: relevant Stan code excerpts

```
transformed data{
    //Calculate "observed" LG parameters from MINOS probabilities
    ⋮
}

parameters{
    real deltam2; //Neutrino oscillation parameters
    real sinsq2theta; //(extracted by "fitting" data to neutrino
                        survival probability model)
    vector[nData] normal_dist; //Pseudo-data modeled using Gaussians
    ⋮
}

transformed parameters{
    ⋮
    Pmumu = 1.-sinsq2theta*square(sin(phase(deltam2, E)));
    Pmumu_sample = Prob + (normal_dist*Prob_err); //Pseudo-data
}
```

The analysis: relevant Stan code excerpts

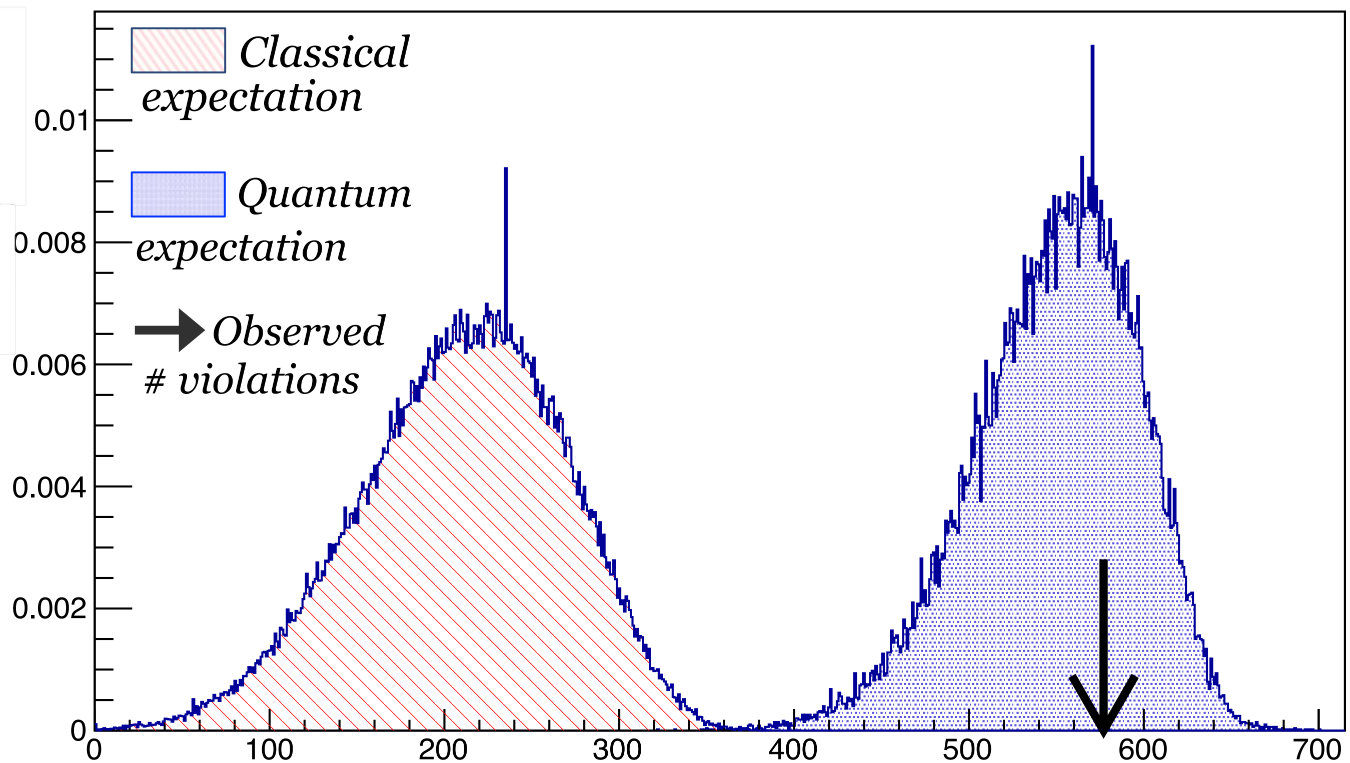
```
transformed parameters{
    ⋮
    //Calculate quantum or classical LG-parameters from pseudo-data,
      depending on "isQuantum."
}

model{
    Prob ~ normal(Pmumu, Prob_err); //"Fit" oscillation parameters
                                      to survival probability data
    normal_dist ~ (0., 1.);
    ⋮
}

generated quantities{
    //Determine # observed K points that violate classical LGI bound
    //Determine # sampled K points (classical or quantum prediction)
      that violate the bound
}
```

The data appear inconsistent with a classical (hidden variable) model



The data appear inconsistent with a classical (hidden variable) model

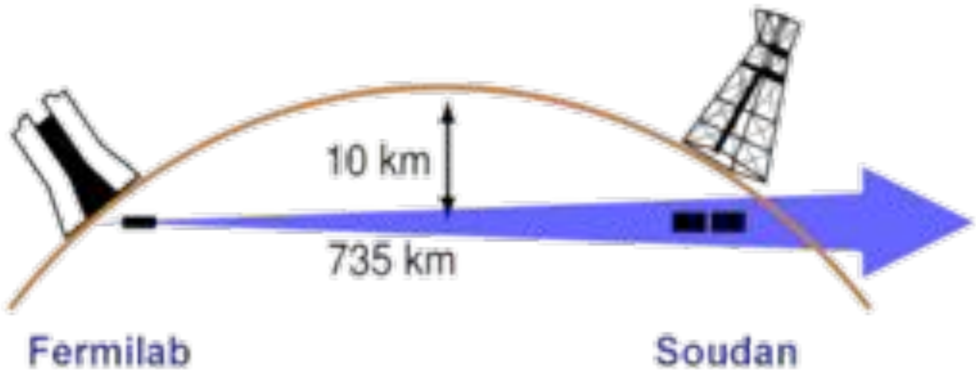
Posterior probability distributions



Number of LGI violations for K_4

Using Stan, we performed the first test of the LGI on a neutrino system

We find that 64 of 82 K_3 values and 577 of 715 K_4 values violate the LGI.



A class of *hidden variable* theories (for a *deterministic* world) is disfavored.

These results provide evidence that the neutrino occupies no definite state until it is measured—and that the "quantum weirdness" of neutrinos prevails across 100s of kilometers.

Stan and Physics:
A symbiotic relationship

Part 1: Testing interpretations of quantum mechanics ✓

Does quantum weirdness exist?

Part 2: Modeling the neutrino mass problem

Can we experimentally determine the neutrino masses?

We employed Stan:

– To generate pseudo-data
– To extract physical parameters from data

Stan and Physics:
A symbiotic relationship

Part 1: Testing interpretations of quantum mechanics ✓

Does quantum weirdness exist?

Part 2: Modeling the neutrino mass problem

Can we experimentally determine the neutrino masses?

We employed Stan:

– To generate pseudo-data
– To extract physical parameters from data

Image: Symmetry Magazine

Morpho: Our python-based wrapper for Stan

Morpho bridges between Stan/PyStan and data input/output.

Features include:

- Loading data & Stan functions
- Loading & outputting to root, hdf5 and R files
- Running Stan diagnostic tests
- Creating plots (histograms, divergence plots, etc.)

The neutrino mass problem:
Probing masses millions of times lighter than the electron's

Unknown: the size and ordering of neutrino masses m_1, m_2, m_3

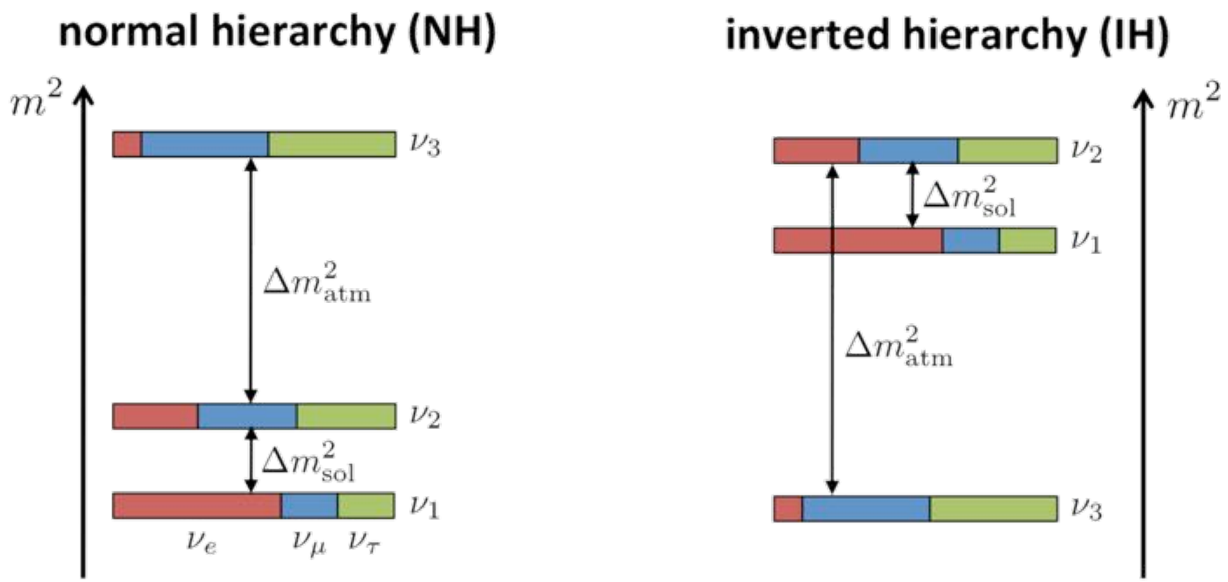
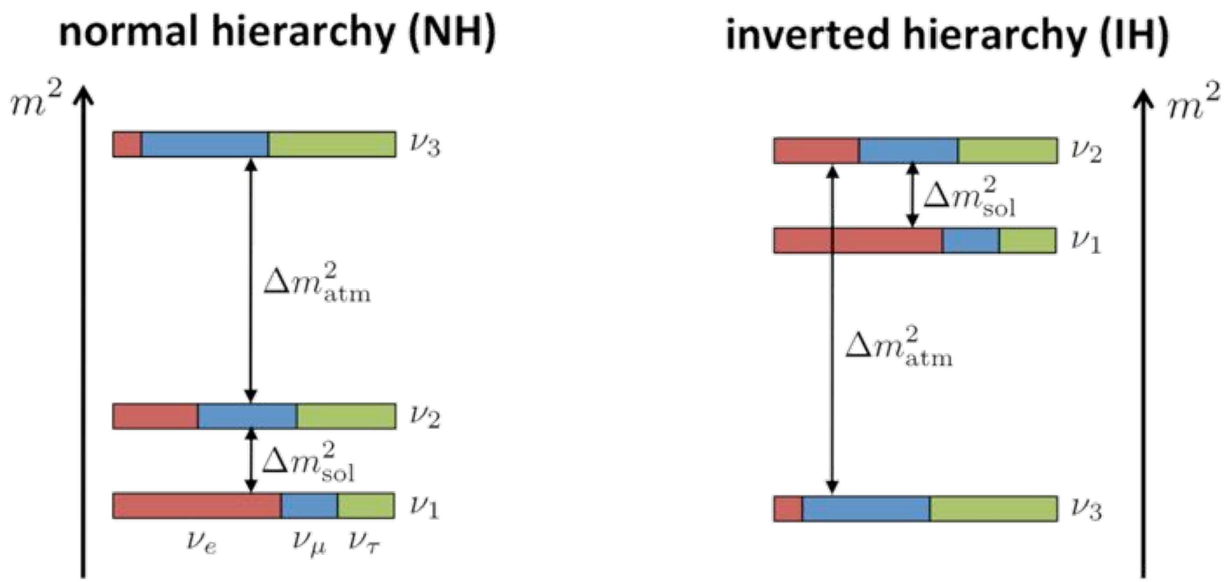


Image: Mainz

The neutrino mass problem:
Probing masses millions of times lighter than the electron's

Unknown: the size and ordering of neutrino masses m_1, m_2, m_3

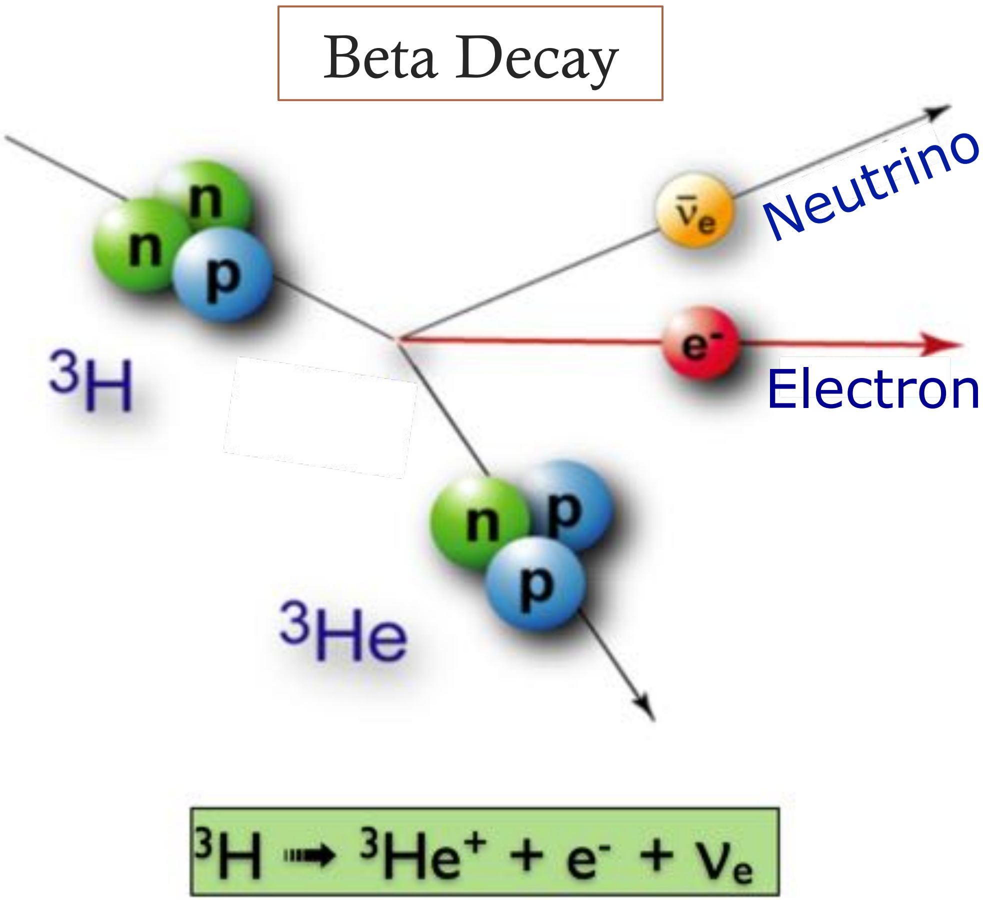


Implications of a mass measurement:

Cosmology (structure formation models); matter-antimatter asymmetry

Image: Mainz

Beta decay: a "direct" approach to measuring neutrino masses

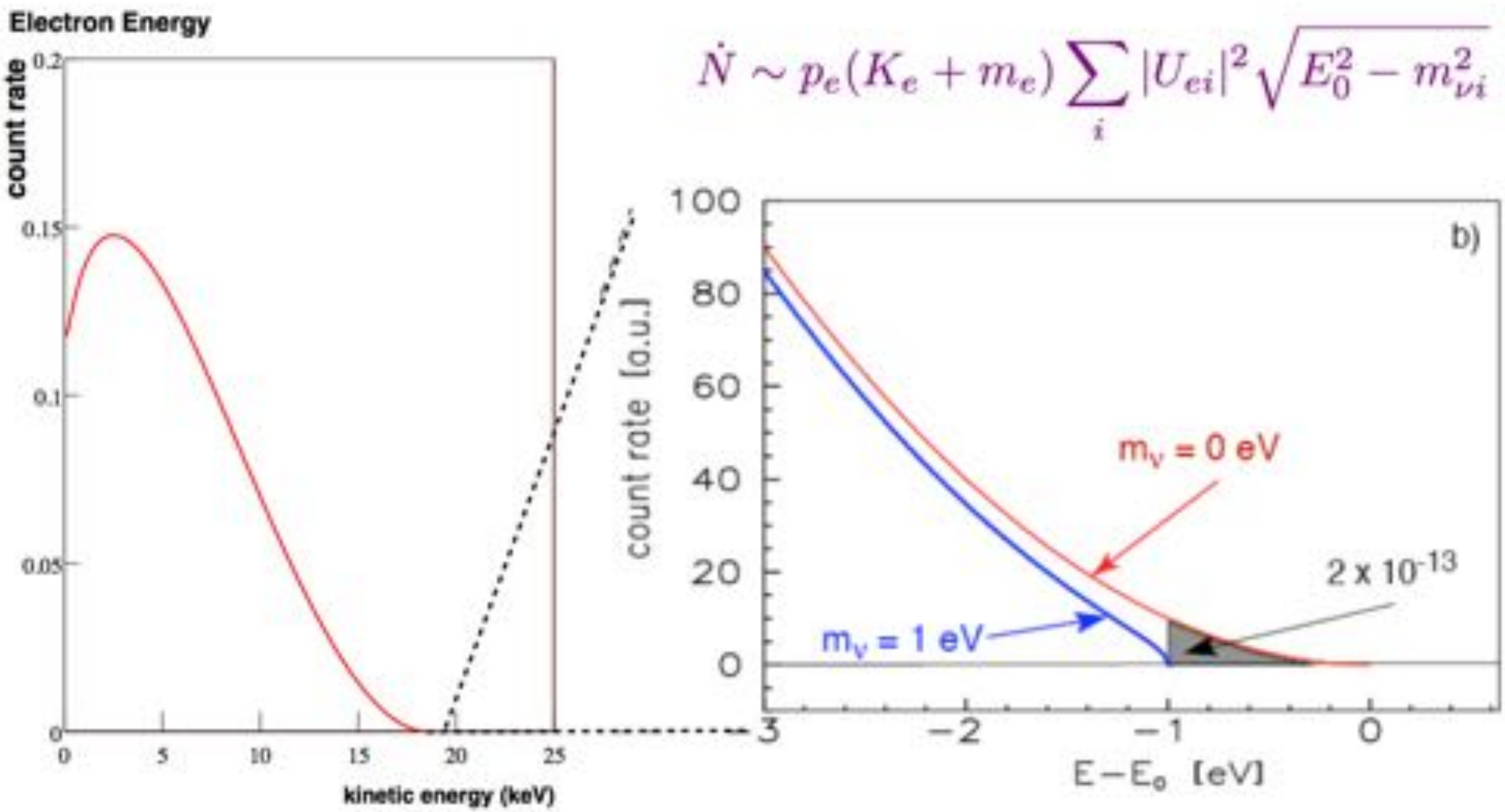


Determine neutrino masses from "*missing energy*"

No model dependence (on cosmology or mass ordering)

Beta decay: a "direct" approach to measuring neutrino masses

$$\dot{N} \sim p_e(K_e + m_e)\sum_i |U_{ei}|^2\sqrt{E_0^2 - m_{\nu i}^2}$$



Plots: Formaggio (2017)

Project 8: The frequency approach

Project 8 uses cyclotron frequencies $\omega(\gamma)$ to extract electron energies.

➡ **Better precision.**

PROJECT 8

$$\omega(\gamma) = \frac{\omega_0}{\gamma} = \frac{eB}{K + m_e}$$

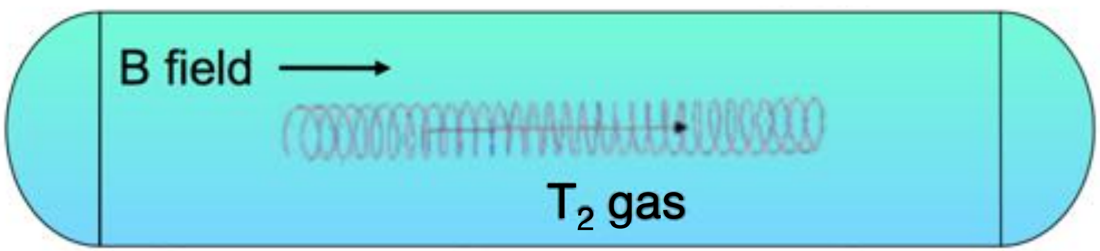
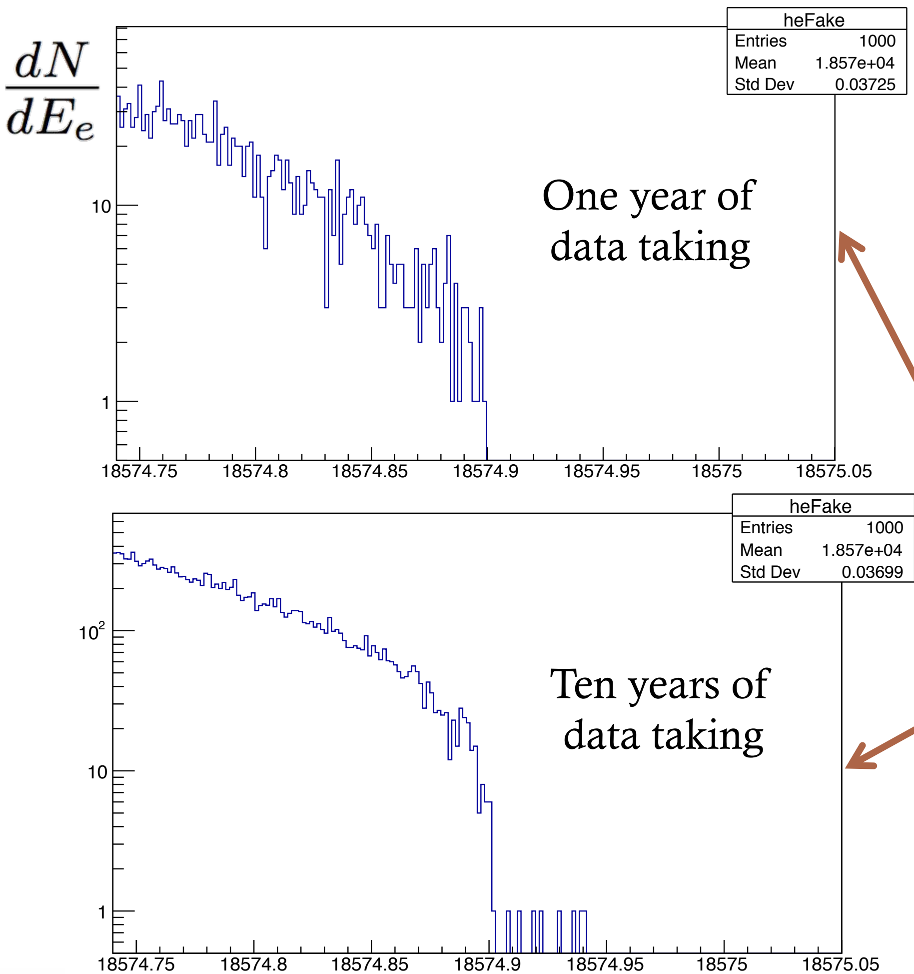


Image: Formaggio (2017)

T_2 beta decay spectra generated in Stan
(*Project 8*-specific analysis)



E_e (eV)

We are working to generate and analyze spectral data for T and T_2.

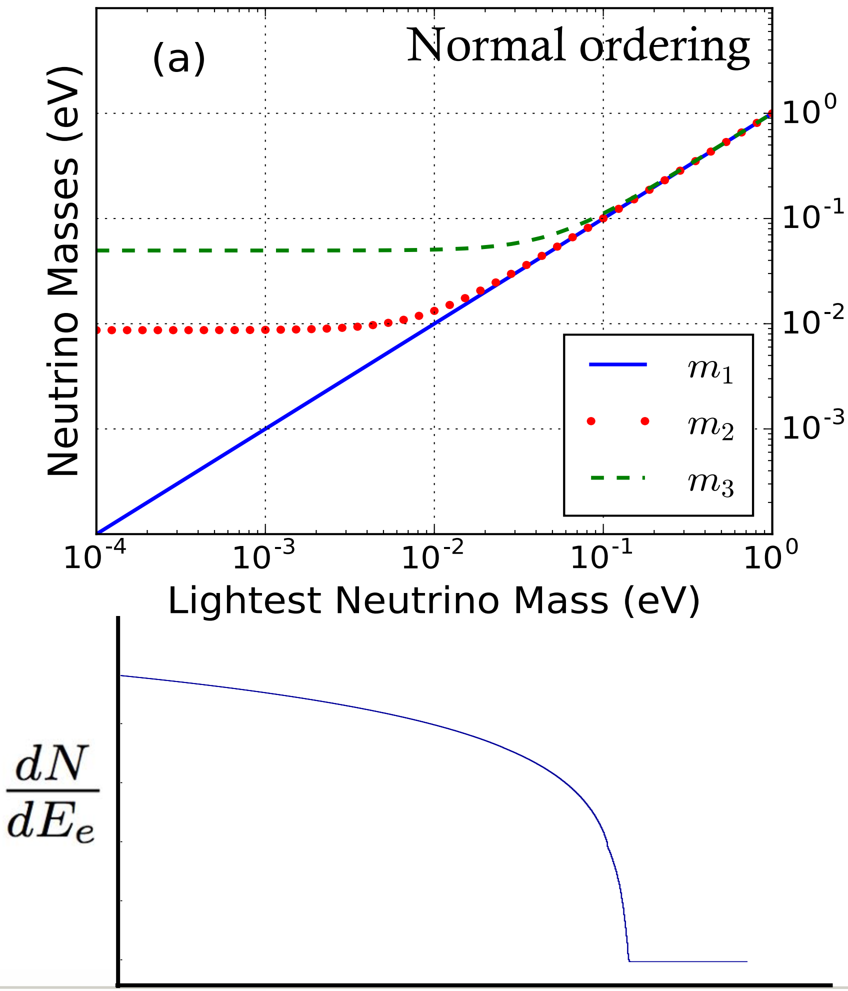
10^{19} tritium atoms (effective volume of 10 m^3)

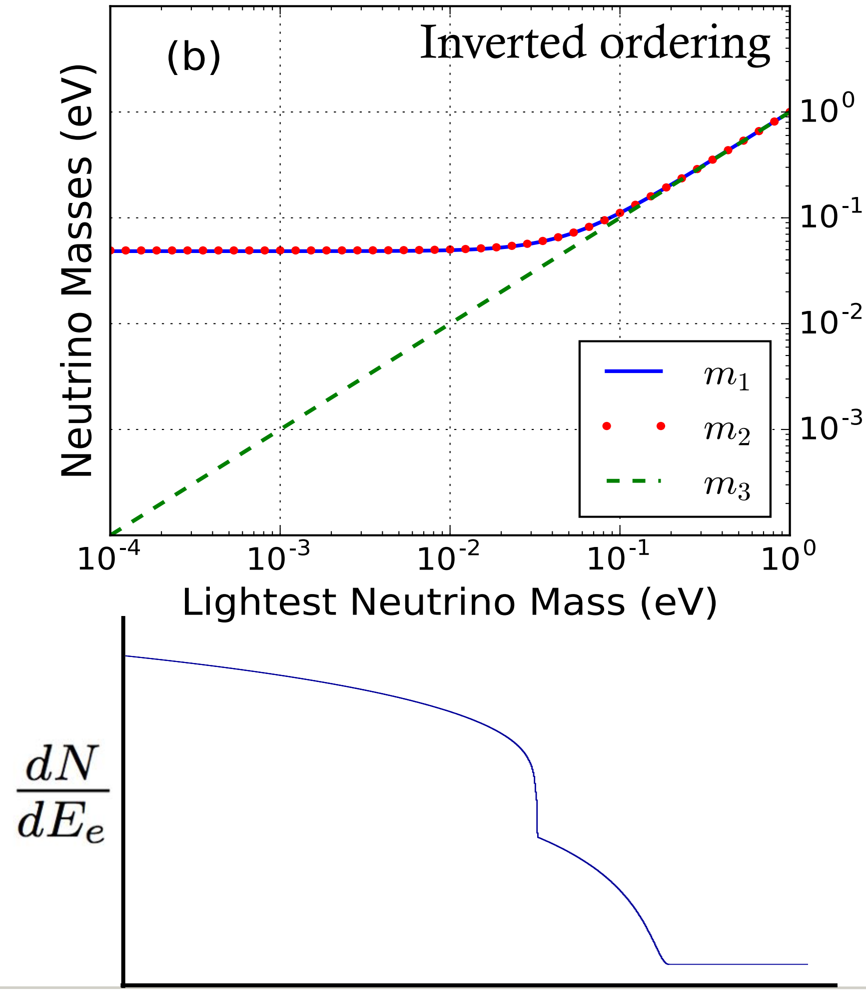
Assuming inverted hierarchy

Plots by Mathieu Guigue (PNNL)

A direct spectral shape analysis accounts for "kinks" near the endpoint:

potential gains from *Project 8*





Incorporating external neutrino data

Incorporate oscillation parameter measurements from other experiments as priors in analysis.

Allow for errors in reported uncertainties $\longrightarrow$ $\boxed{\begin{array}{l}\theta \sim (\mu_\theta, \sigma_\theta) \\ \sigma_\theta \sim (\mu_{\sigma_\theta}, \tau)\end{array}}$

Experiment	θ_{12}	θ_{13}	Δm^2_{32}	Δm^2_{21}	Δm^2_{13}
KamLAND (2013)	$\tan^2\theta_{12}$	$\sin^2\theta_{13}$	×	✓	×
SNO (2011)	$\tan^2\theta_{12}$	×	×	✓	×
RENO (2014)	×	$\sin^2 2\theta_{13}$	×	×	×
Double Chooz (2014)	×	$\sin^2 2\theta_{13}$	×	×	×
T2K (2014-15)	×	$\sin^2 2\theta_{13}$, h-dep	normal	×	inverted
Noνa (2016)	×	×	✓	×	×
MINOS (2014)	×	×	✓	×	×
Daya Bay (2014)	×	$\sin^2 2\theta_{13}$	✓	×	×
IceCube (2015)	×	×	✓	×	×

The challenge of determining the likelihood of each mass ordering in Stan

Bimodality problem

- Infinite time → probability that Markov chains converge to one solution reflects probability that data is consistent with a mass ordering
- But our time is not infinite …

Possible solutions

1. Hierarchical mixing:

$$\Big[\lambda\pi(D|\vec{m}, \text{normal}) + (1-\lambda)\pi(D|\vec{m}, \text{inverted})\Big]\pi(\ldots)$$

2. Adiabatic Monte Carlo:
   Employs geometries from equilibrium thermo

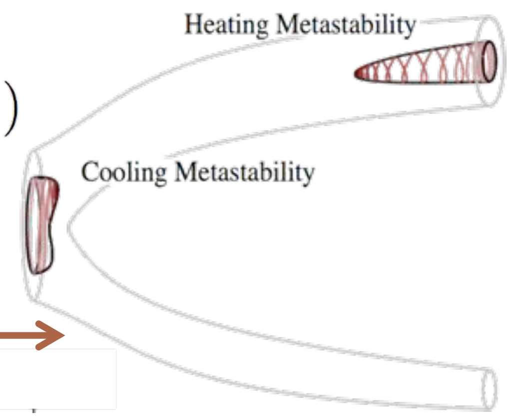


Image: "Adiabatic Monte Carlo," Betancourt

An approximate model of the tritium beta decay spectrum

- We map a fixed window in frequency space to energy space. This "rounds the edges" around the window.

- Challenge: the spectrum is a *convolution*.
Expand in neutrino mass to 1st order $\longrightarrow$ analytic approximation:

$$\mathcal{P}(K) \;=\; ((Q-K)^2 - \frac{1}{2}m^2)\cdot\Theta(Q-K-a)\Theta(b-(Q-K))$$

- Convolution with a normal distribution:

$$\mathcal{F}(K) \;=\; \int P(K')\cdot\mathcal{N}(K'|K,\sigma)dK'$$

An approximate model of the tritium beta decay spectrum

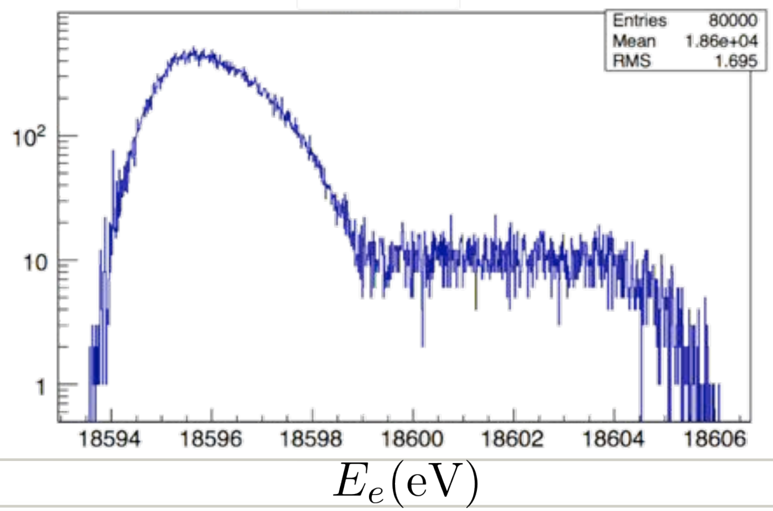
Convolution with a normal distribution:

$$\mathcal{F}(K) = \int P(K') \cdot \mathcal{N}(K'|K,\sigma) dK'$$

$$\mathcal{F}(K|Q,m,\sigma,b) = N(m,b) \cdot (\gamma(Q-K,m,\sigma,m) - \gamma(Q-K,m,\sigma,b))$$

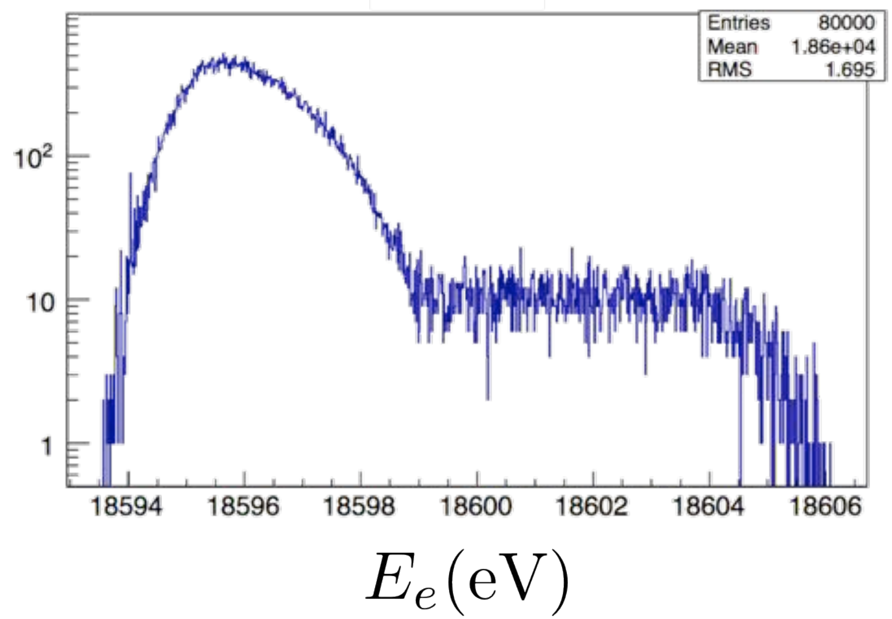
$$\gamma(z,m,\sigma,a) = (a+z)\sigma^2\mathcal{N}(z|a,\sigma) + \frac{1}{2}\left(-\frac{m^2}{2} + z^2 + \sigma^2\right) \mathrm{Erfc}\left[\frac{a-z}{\sqrt{2}\sigma}\right]$$

This is smooth and normalizable.
Now implemented in morpho.



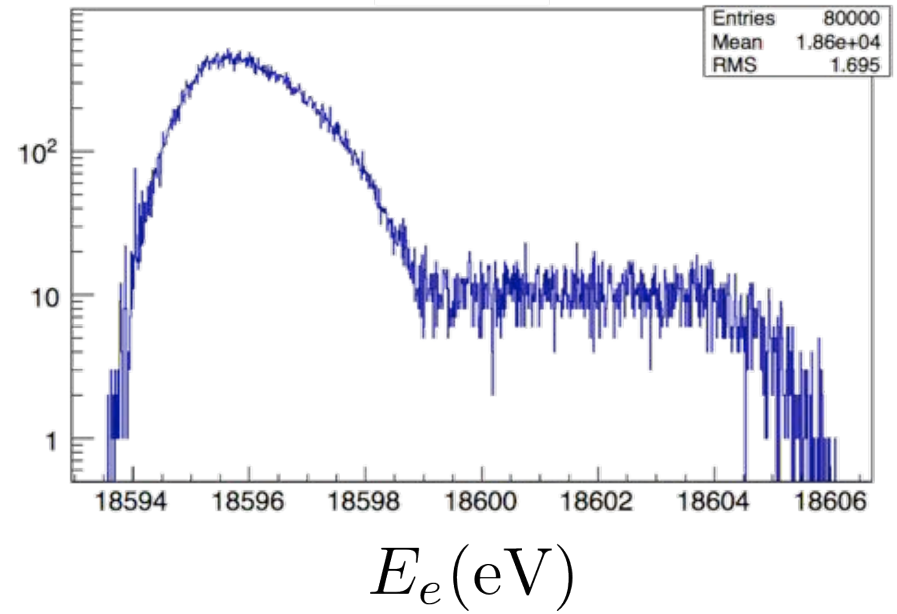
A test of the approximated beta decay spectrum in Stan

- **Energy window:** 2 eV
- **Neutrino mass:** 10 meV
- **Endpoint** (Q): 18600 eV
  Assume 0.2 eV uncertainty, from external constraint (normal prior)
- **Signal fraction:** 90%
- **Detector resolution**: 50 meV
  Assume 10% uncertainty, from external constraint (normal prior)



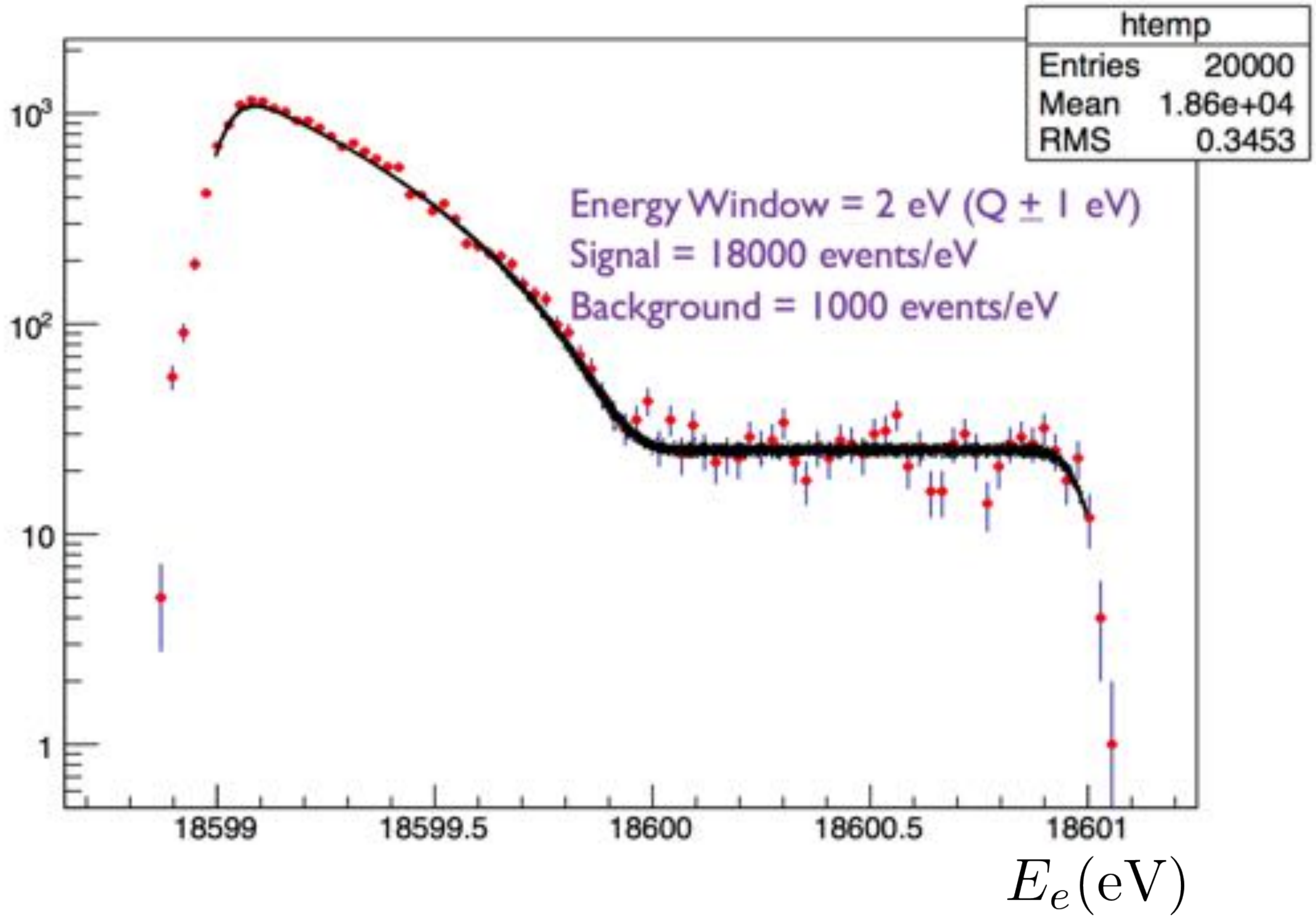
A test of the approximated beta decay spectrum in Stan

- **Energy window:** 2 eV
- **Neutrino mass:** 10 meV
- **Endpoint** (*Q*): 18600 eV
  Assume 0.2 eV uncertainty, from external constraint (normal prior)
- **Signal fraction:** 90%
- **Detector resolution**: 50 meV
  Assume 10% uncertainty, from external constraint (normal prior)

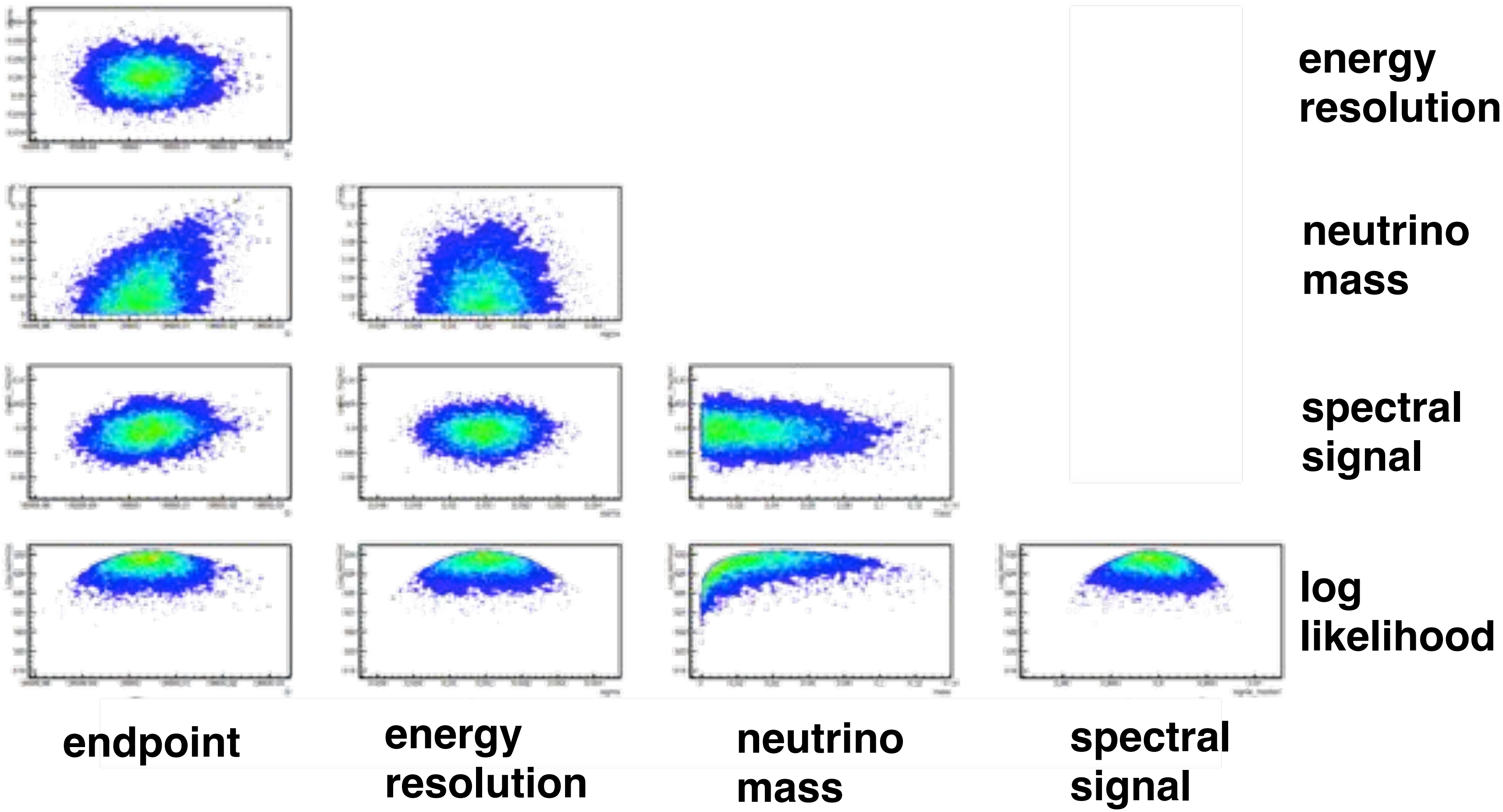


20,000 events generated (in ~1s).
Spectrum analyzed with 4 chains (in ~1 hr).

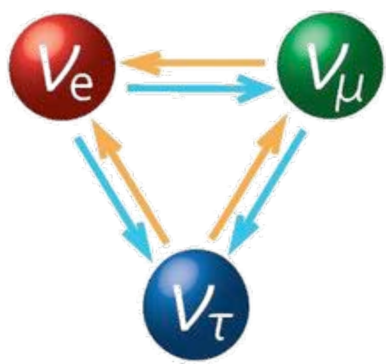
A test of the approximated beta decay spectrum in Stan

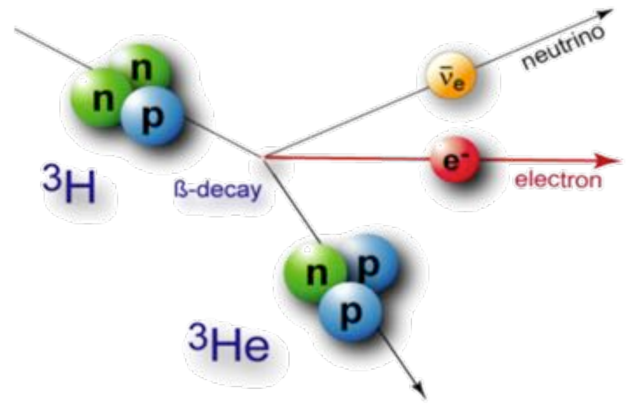


Parameters distributions from analysis appear to converge around inputted values



Advancing physics research with Stan: Returning to our observations





Stan is a powerful tool for physicists because it enables:

- Modeling complex systems

  Can capitalize on detailed experimental knowledge

- Extracting physical parameters from data

  Precludes need to create own "fitters"

Thank You