The Basic Steps to a Physics Analysis
(Using $B^{\pm} \to K^{\pm}\tau^{+}\tau^{-}$ as an example)

Chris Ketter
University of Hawaii at Manoa
Belle II KLM Group

1) Obtain signal Monte Carlo
2) Study the physical process (What's unique about the decay mode?)
3) Optimize signal selection (tune cuts on pID, kinematics, etc.)
4) Reject background (additional cuts, MVA techniques)
5) Make a measurement (cut & count / functional fit / template fit)
6) Validation (measure a control mode)
7) Evaluate systematics
8) Unblind
9) Publish!

Colors
- Completed and/or in progress
- Outstanding

Signal Monte Carlo

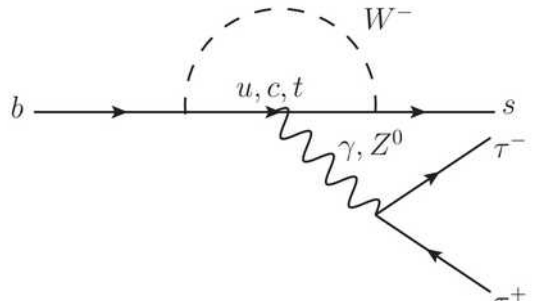
- A blind analysis means completing whole analysis on Monte Carlo (MC) data before looking at real data
- For Belle analyses, need to generate $B^+ \to K^+ \tau^+ \tau^-$ MC (for Belle II, one should consult data production group):
  - Want separate MC data sets for all τ channels under consideration, e.g. $\tau \to e\nu\bar{\nu}$, $\tau \to \mu\nu\bar{\nu}$, $\tau \to \pi\nu$
  - Generate MC decay tables (.gen files) with `evtgen` (using mcproduzh pkg.)
    - Example decay.dec file shown for $B^{\pm} \to K^{\pm}[\tau^+ \to e^+\nu_e\bar{\nu}_\tau][\tau^- \to e^-\nu_\tau\bar{\nu}_e]$ →
  - Simulate detector response (.mdst files) with `Geant 3` (Belle II uses `Geant 4`)

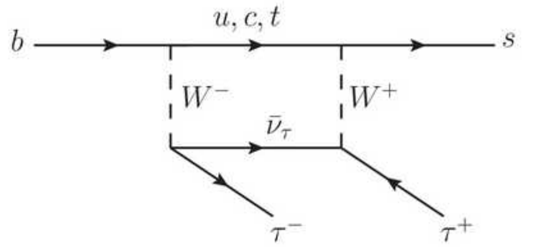
```
# Define Aliases
Alias MyB+ B+
Alias MyB- B-
Alias MyTau+ tau+
Alias MyTau- tau-

yesPhotos  # Turn on PHOTOS for all decays

#### BF ###   ###### Daughters #######     # Generator #
Decay Upsilon(4S)
0.500000000   B+      MyB-                         VSS;
0.500000000   MyB+    B-                           VSS;
Enddecay

# Signal-side decay
Decay MyB-
1.000000000   K-      MyTau+      MyTau-       BTOSLLBALL;
Enddecay
CDecay MyB+

Decay MyTau-
1.000000000   e-      anti-nu_e   nu_tau       TAULNUNU;
Enddecay
CDecay MyTau+

End
```

Def: tag B meson = the other B meson that is not your signal B meson. (variations incl, hadronic, semileptonic, inclusive)

Physical Process

- Final state has 2-4 neutrinos, so missing mass is a hallmark of this decay mode
- The signal kaon is not missing any mass, and it's momentum is anti-correlated with the momentum of the $\tau^+\tau^-$ system which cannot be reconstructed
- Theoretical branching fraction very small
  - Could be inflated with different NP scenarios
  - Decided to use inclusive tagging method to maximize statistics (at the expense of resolution)
- Would like to fit 2D distribution of missing mass^2 vs. transverse momentum of the kaon
  - Therefore, we don't want to cut on these (or variables highly correlated with these), nor do we want to use them in any MVA training
- Also, we know $B^{\pm} \to K^{\pm}[J/\Psi \to \ell^+\ell^-]$ has same final state as some of the 1-prong τ modes, so we can use this as a control mode to validate the our procedure

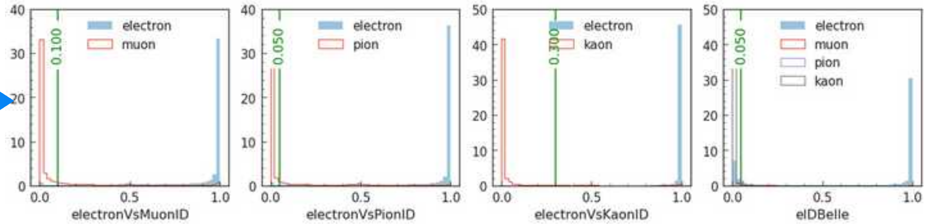


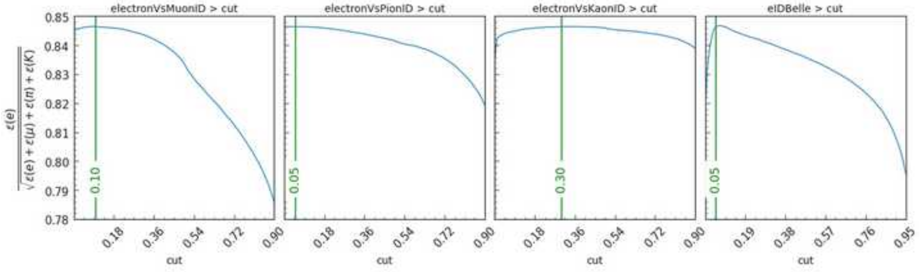


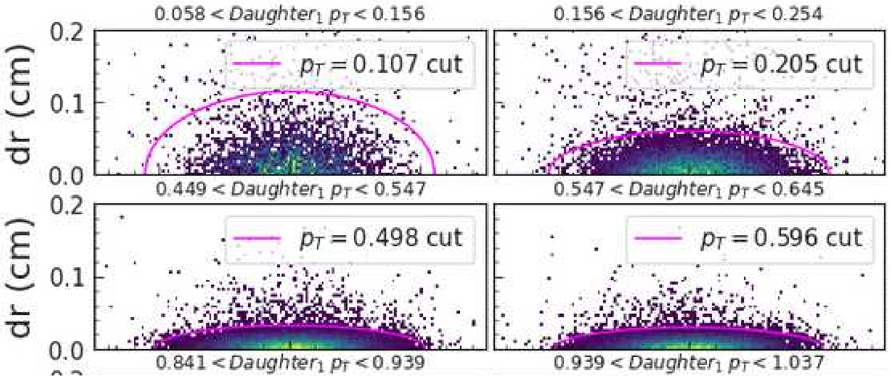
$$\begin{aligned}
N_{theory} &= N_{B^+B^-} \cdot Br(B \to K\,\tau\,\tau) \cdot Br(\tau \to e\nu\bar{\nu}) \cdot Br(\tau \to e\bar{\nu}\nu) \\
&= 384735950 \cdot (1.61 \cdot 10^{-7}) \cdot 0.1779 \cdot 0.1779 \cdot 2 \\
&\approx 4 \\
N_{Br=3.0\cdot10^{-4}} &= 384735950 \cdot (3 \cdot 10^{-4}) \cdot 0.1779 \cdot 0.1779 \cdot 2 \\
&\approx 7304 \\
N_{reco-expected} &= N_{Br=3.0e-4} \cdot \epsilon_{reco} \\
&= 7304 \cdot 0.0769 \\
&\approx 562
\end{aligned}$$

Signal Selection

- Study signal MC --- look at distributions of things like pT, pID, and impact parameters of final state particles and establish cuts
  - Use these cuts to build final-state particle lists (`ma.fillParticleList`)
- Skimming --- discard events that don't pass skimming criteria (`ma.applyEventCuts`)
  - Event shape should be consistent with an Y(4S) decay (`ma.buildEventShape`)
  - In the case of inclusive tagging, the tag side should be free of leptons (`countInList`)
  - I require that the whole event has the exact number of electrons and or muons for each of my decay channels
- Reconstruct Decay --- use every combinatorial way (candidates) to build the specified decay chain for each event (`ma.reconstructDecay`)
  - Can reduce number of incorrect candidates here by imposing additional cuts (e.g. `Mbc`, `deltaE`)
- Use the rest of the tracks and clusters to build my inclusive tag B meson (`ma.buildRestOfEvent`)
- Make additional cuts on the quality of the tag B meson (`ma.applyCuts`)
  - Can help eliminate some of the incorrect candidates
- Perform vertex fits for both the signal B and tag B mesons (`vtx.treeFit`, `vtx.TagV`)
- From all the candidates in the event, choose one (`ma.rankByHighest`)
  - I choose the candidate with the highest p-value combination of the signal and tag side fitted p-values

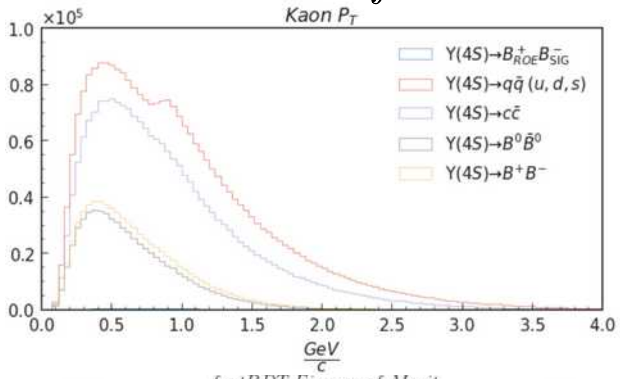


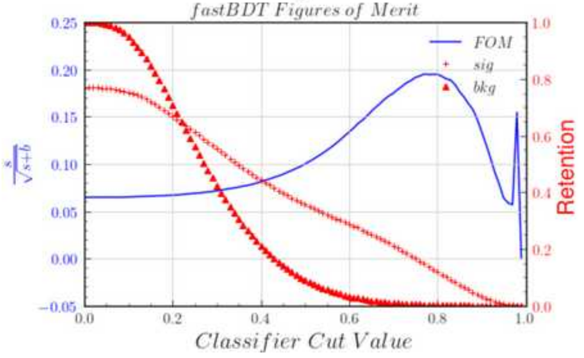


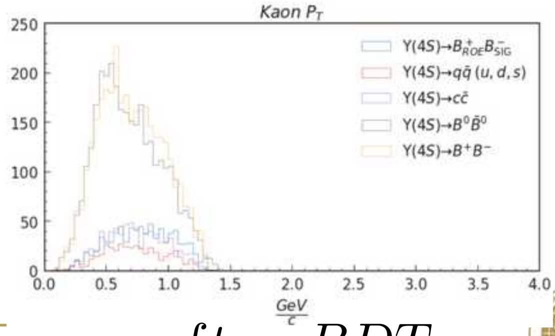


Background Rejection

before BDT







after BDT

- Were studying e+e- → Y(4S), but we can also have $e^+e^- \to q\bar{q}\ (q = u, d, c, s),\ \ \ell^+\ell^-,\ \ \gamma\gamma$
- Further, we must consider $\Upsilon(4S) \to B\bar{B} \to X$
- We run our steering script on both signal and various background MC types
- First, we tune our loose cuts to reject as much background as possible
- To further improve background rejection, we can use machine learning techniques (I'm using `fastBDT`)
  - Signal and BG nTuples are used train a boosted-decision tree
  - I use one BDT to reject continuum $e^+e^- \to q\bar{q}\ (q = u, d, c, s)$ and another BDT for all other charged/neutral B meson decays
  - Finally, the cut value for each BDT output classifier is chosen using a figure of merit, e.g. $S/\sqrt{S+B}$

Fitting Unknown Distributions

- Often times, we may not have a probability distribution function which accurately describes our data
  - In this case we can fit to a template
- With `pyhf` (Histogram Factory for python) we can construct a model of any binned data and fit the model to independent data
- It's a maximum likelihood estimator based on pdf's that assume underlying Poisson statistics
- The fit (maximum likelihood estimate) finds the values of signal strength, μ, and background bin contents, θ, which maximize L(μ, $\boldsymbol{\theta}$)
- Significance is measured by assuming a null hypothesis and looking for an excess (significance = sqrt(q_0))
- If significance is < 5σ (typical discovery threshold), an upper limit can be measured by scanning over different signal strengths and finding the point where the cdf(q_μ|μ) = chosen exclusion threshold

$$L(\mu, \boldsymbol{\theta}) = \prod_{j=1}^{N} \frac{(\mu s_j + b_j)^{n_j}}{n_j!} e^{-(\mu s_j + b_j)} \prod_{k=1}^{M} \frac{u_k^{m_k}}{m_k!} e^{-u_k} .$$

$$\lambda(\mu) = \frac{L(\mu, \hat{\hat{\boldsymbol{\theta}}})}{L(\hat{\mu}, \hat{\boldsymbol{\theta}})}$$

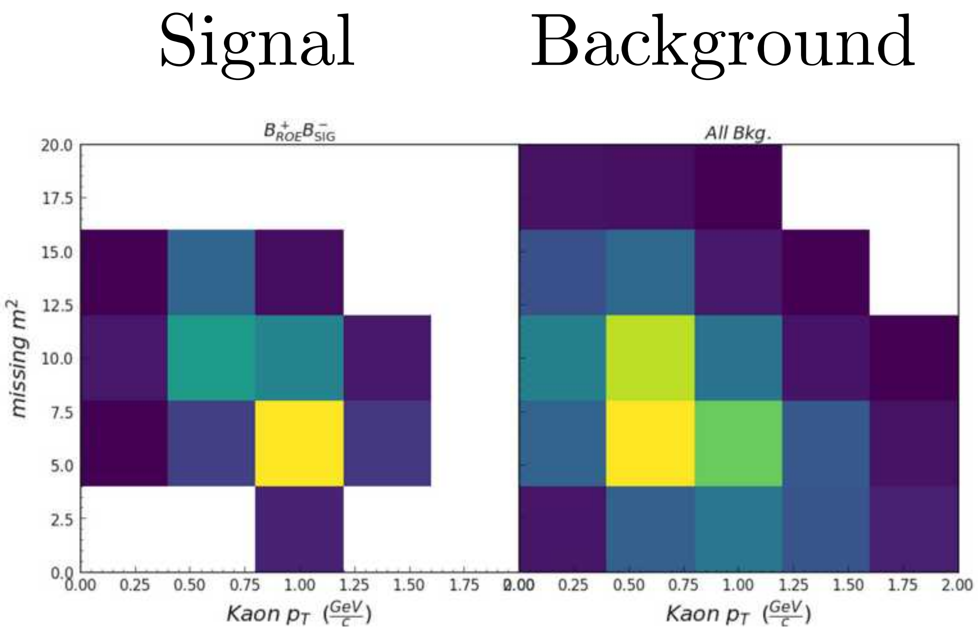
$$q_0 = \begin{cases} -2\ln\lambda(0) & \hat{\mu} \geq 0 , \\ 0 & \hat{\mu} < 0 , \end{cases}$$

$$q_\mu = \begin{cases} -2\ln\lambda(\mu) & \hat{\mu} \leq \mu \\ 0 & \hat{\mu} > \mu \end{cases}$$

$$p_\mu = \int_{q_{\mu,\mathrm{obs}}}^{\infty} f(q_\mu|\mu)\, dq_\mu$$

My 2D Fits

- Right: my 1st attempt at a template for signal and background of kaon p_T vs. missing m^2
  - To build the pyhf model, the bin counts of the 2D histograms (signal and background) are simply flattened into a 1D array and normalized by the expected yield
  - Bins without signal automatically become side bands and help constrain the background amplitudes in the signal region
- One can combine multiple decay channels into one simultaneous fit --- this one is just one tau channel
- In this 1st attempt fitting MC data, with an assumed branching fraction 3 x 10^{-4}, significance was < 5σ
- A scan for the required signal strength reach a 10% exclusion level already gives an upper limit of 1.8 x 10^{-3} at 90% C.L.



Remaining Steps (Future Work for Me)

- Validate analysis on a control mode
  - Ideally something well studied
  - Can unblind control mode for validation
- Study systematics
  - What are the effects of systematics [pre-selection cuts, signal & background modeling, BDT, ...] on my measurement [significance, upper limit]?
- Unblind
  - Run my reconstruction on real Belle data
  - Partial unblinding?
    - First check side bands where no signal is expected?
- Publish!

Summary

- I've tried to lay out the basic steps to performing a HEP analysis
  - Steps may vary for different types of analyses
- With limited time for this talk, I've omitted a description of machine-learning techniques
  - For more information on this, I recommend Simon Wehle's presentation at the 2019 BNL Workshop https://indico.bnl.gov/event/5655/
- I've also omitted a discussion on inclusive ROE tagging as this will be covered in an upcoming talk by Boyang Zhang
- I have tried to introduce you to the popular fitting package (`pyhf / histogram factory`) used in HEP analyses so you may have an idea of how to perform a template fit, calculate significance, and determine upper limits

References

- The outline of this talk was heavily influenced by Michael DeNuccio's talk, *Search for Axion-Like Particles produced in e+ e- collisions at Belle II*, shown at the 2020 B2SW

- Heinrich et al., (2021). pyhf: pure-Python implementation of HistFactory statistical models. Journal of Open Source Software, 6(58), 2823, https://doi.org/10.21105/joss.02823
- Glen Cowan, Kyle Cranmer, Eilam Gross, Ofer Vitells, *Asymptotic formulae for likelihood-based tests of new physics*, arXiv:1007.1727