

CLUSTERING FOR LARGE-SCALE REINTERPRETATION OF NEW PHYSICS SEARCHES AT THE LHC

Judita Mamužić
SMASH Research Fellow
Jožef Stefan Institute, Slovenia
20. October 2026

PRELIMINARY

ATLAS EXPERIMENT

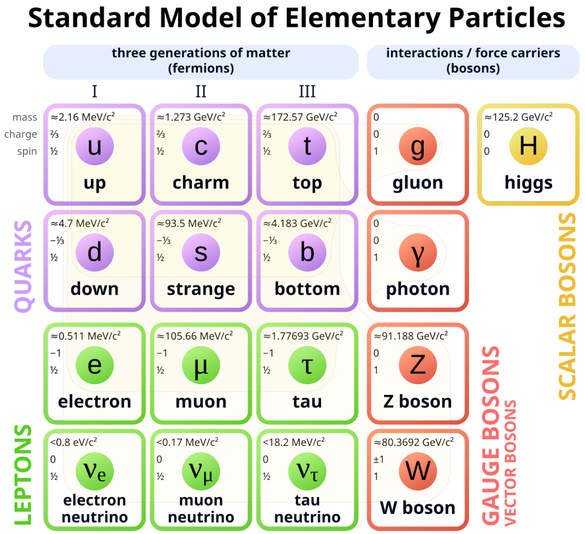
Jožef Stefan Institute

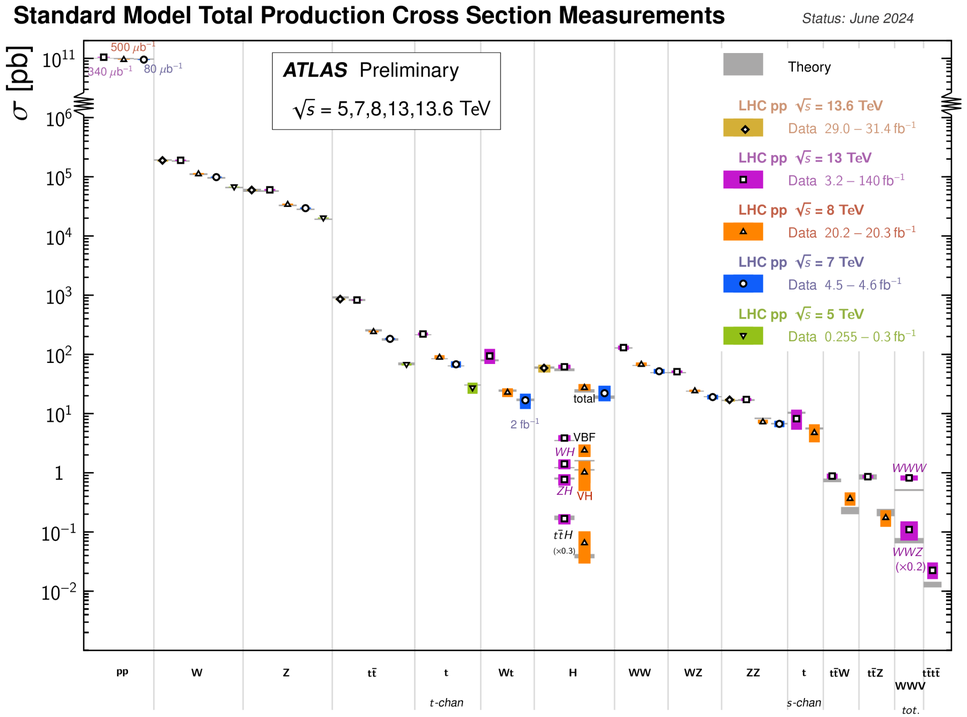
I FEEL SLOVENIA

Co-funded by the European Union

SMASH
machine learning for science and humanities postdoctoral program

STANDARD MODEL AND PHYSICS BSM





ATL-PHYS-PUB-2024-011

Cosmic microwave background measurements

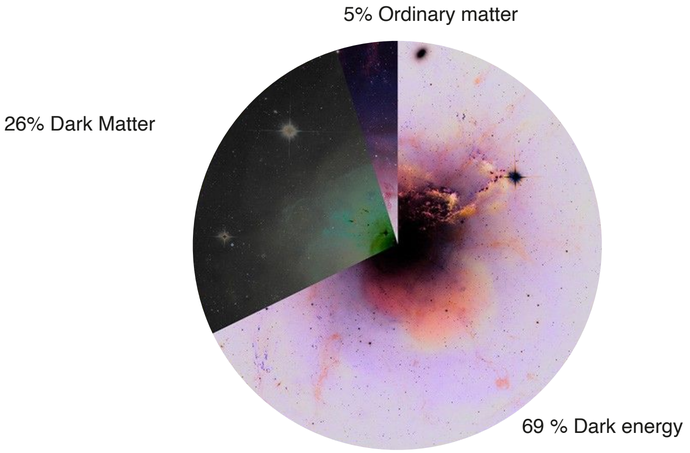
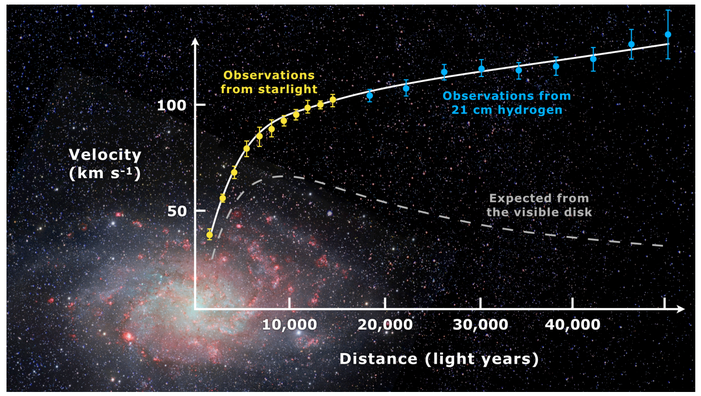


Image modified from NASA Dark Matter

Galactic rotation curves



doi:10.1046/j.1365-8711.2000.03075.x

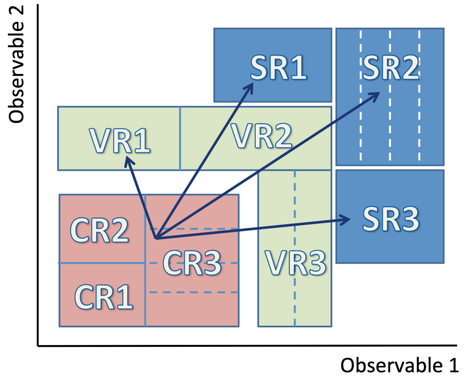
- **Standard Model** describes well the building blocks of matter (quarks and leptons), and their interactions (electromagnetic, weak and strong), with force carriers (photons, gluons and W/Z bosons), acquiring mass via the Braun-Englert-Higgs mechanism
- Good agreement of experimental results with the theory prediction

- A number of phenomena cannot be explained by the SM
- Discrepancies can be explained by the existence of physics **Beyond the Standard Model** (e.g. Dark Matter candidate)

LHC BSM SEARCHES STRATEGY

Strategy

- No favoured **theory** or **mass** of new physics
- Test **any** incarnation of new physics at the LHC
- Impossible to test all models and the parameter space, priority given to feasable searches with best theoretical motivation



DOI:10.1088/1742-6596/664/7/072004

Statistical interpretation

Model independent

$$P(D \mid B)$$

How significant is the excess?

- Rule of thumb in BSM:
  - ~ 3-10 σ SD sign of new physics
  - ~ 2 σ SD: small excess

Model dependent

$$P(D \mid S+B)$$

$$CL_S = \frac{p_{S+B}}{1-p_B}$$

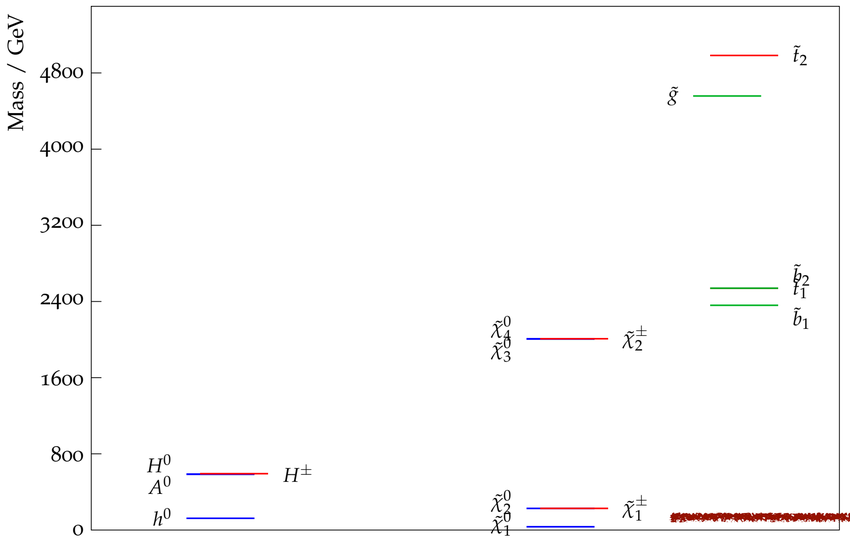
Is the model excluded?

- Model excluded at 95% CL for CLs_obs < 0.05 (exclusion).
- Analyses optimised to maximise parameter space with CLs_exp < 0.05 (sensitivity).

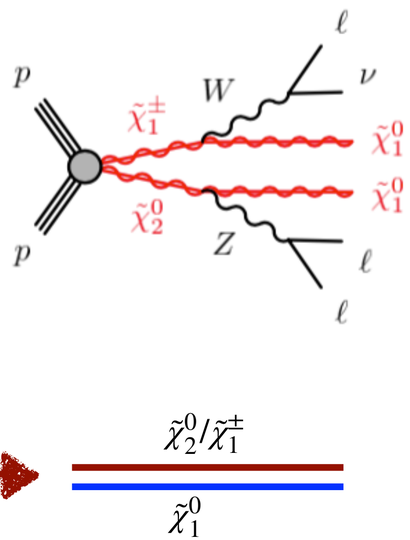
arXiv:1105.3166

Analysis optimization

Full model

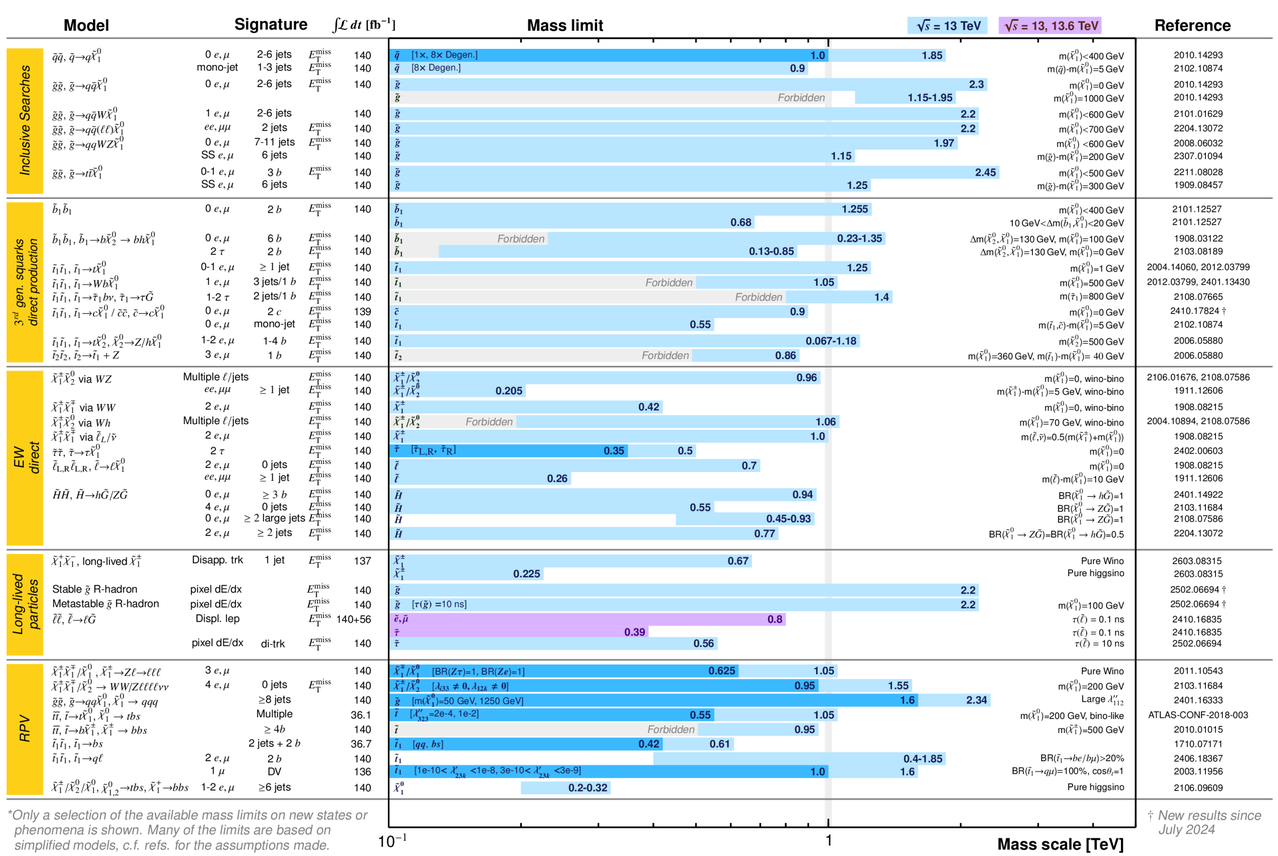


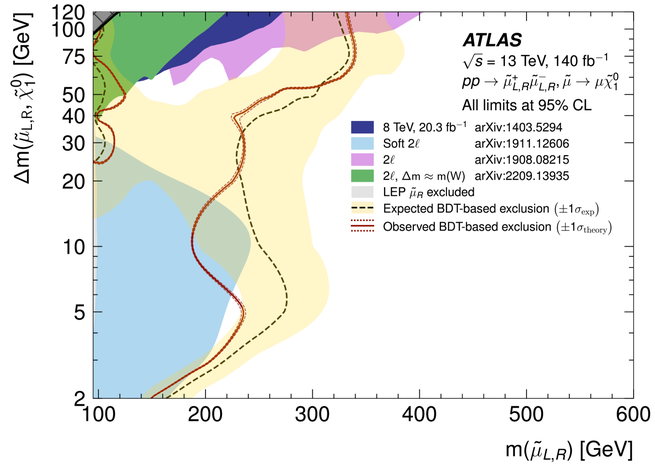
Simplified model



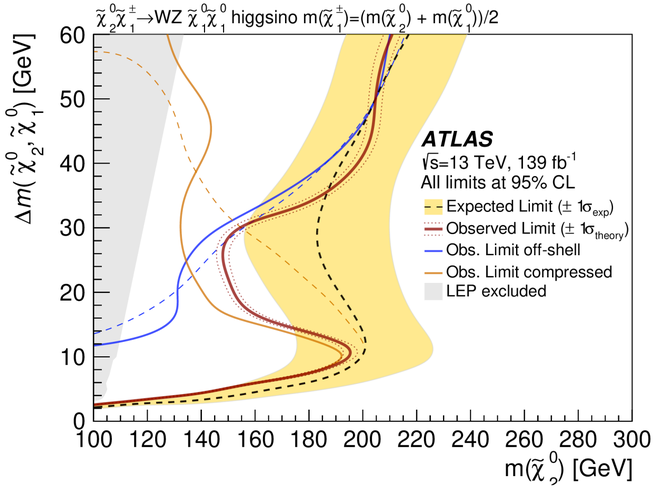
STATUS OF LHC BSM SEARCHES

ATLAS SUSY Summary Plots

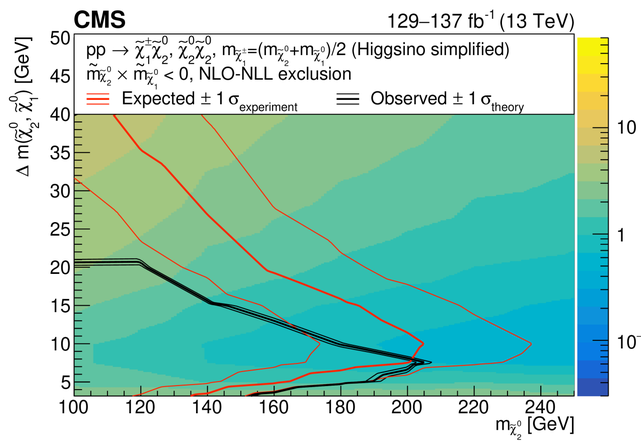




HMBS-2024-64, JHEP (2025)



ATLAS SUSY-2018-16, PRD (2020)
ATLAS SUSY-2019-09, EPJC (2021)

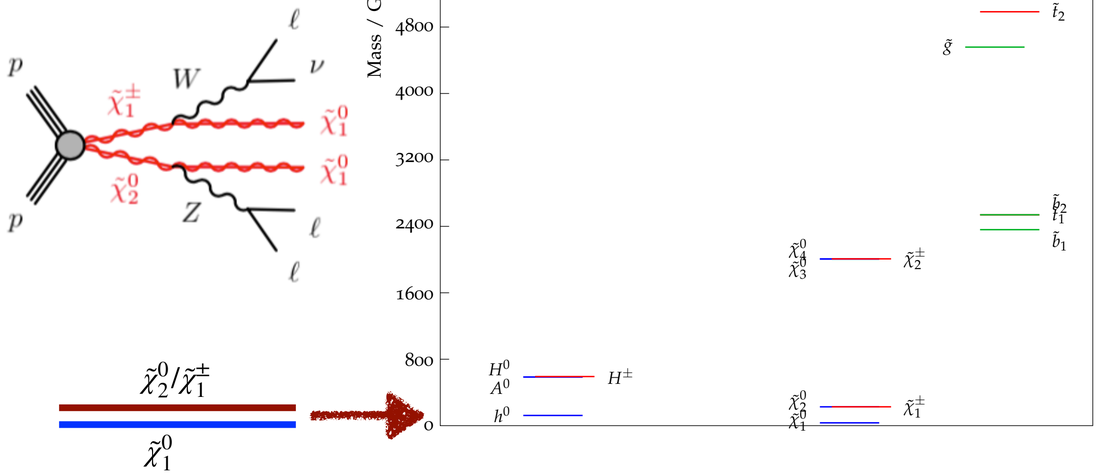


CMS SUS-18-004

- In BSM searches in ATLAS or CMS no significant sign of new physics (> 3 σ SD) was found to date 😭, exclusion limits set (mapping of excluded paramter space)
- Few analyses have observed small excesses ~2 σ SD, LHC BSM WG 😀
- Model-independent 2 σ SD excesses should be considered directly!
- Not all BSM models were tested at 3 σ level, discovery at LHC is still viable!

LARGE-SCALE REINTERPRETATION

Simplified model

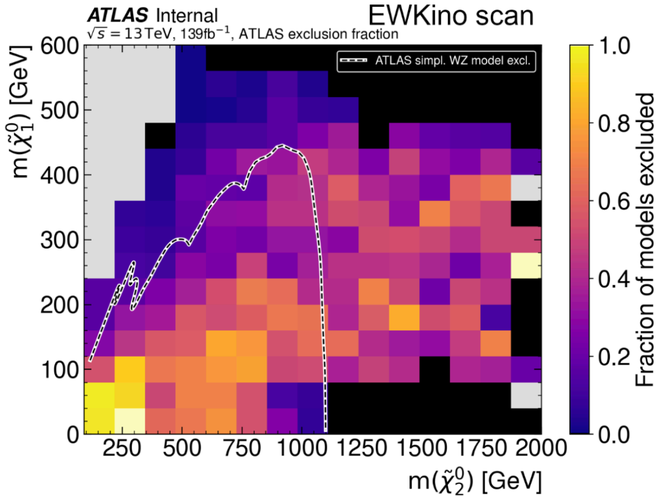


pMSSM model

pMSSM-19 scan

Parameter	min	max	Note
$M_{\tilde{L}_1}$ $(=M_{\tilde{L}_2})$	10 TeV	10 TeV	Left-handed slepton (first two gens.) mass
$M_{\tilde{e}_1}$ $(=M_{\tilde{e}_2})$	10 TeV	10 TeV	Right-handed slepton (first two gens.) mass
$M_{\tilde{L}_3}$	10 TeV	10 TeV	Left-handed stau doublet mass
$M_{\tilde{e}_3}$	10 TeV	10 TeV	Right-handed stau mass
$M_{\tilde{Q}_1}$ $(=M_{\tilde{Q}_2})$	10 TeV	10 TeV	Left-handed squark (first two gens.) mass
$M_{\tilde{u}_1}$ $(=M_{\tilde{u}_2})$	10 TeV	10 TeV	Right-handed up-type squark (first two gens.) mass
$M_{\tilde{d}_1}$ $(=M_{\tilde{d}_2})$	10 TeV	10 TeV	Right-handed down-type squark (first two gens.) mass
$M_{\tilde{Q}_3}$	2 TeV	5 TeV	Left-handed squark (third gen.) mass
$M_{\tilde{u}_3}$	2 TeV	5 TeV	Right-handed top squark mass
$M_{\tilde{d}_3}$	2 TeV	5 TeV	Right-handed bottom squark mass
M_1	-2 TeV	2 TeV	Bino mass parameter
M_2	-2 TeV	2 TeV	Wino mass parameter
μ	-2 TeV	2 TeV	Bilinear Higgs mass parameter
M_3	1 TeV	5 TeV	Gluino mass parameter
A_t	-8 TeV	8 TeV	Trilinear top coupling
A_b	-2 TeV	2 TeV	Trilinear bottom coupling
A_τ	-2 TeV	2 TeV	Trilinear τ lepton coupling
M_A	0 TeV	5 TeV	Pseudoscalar Higgs boson mass
$\tan\beta$	1	60	Ratio of the Higgs vacuum expectation values

ATLAS pMSSM EWK, 20k models

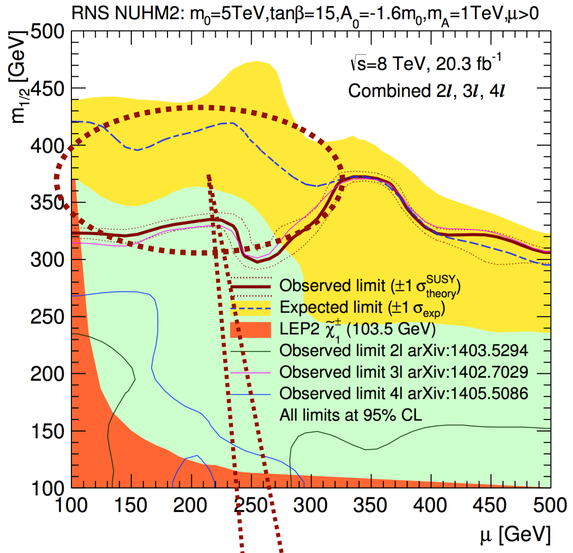


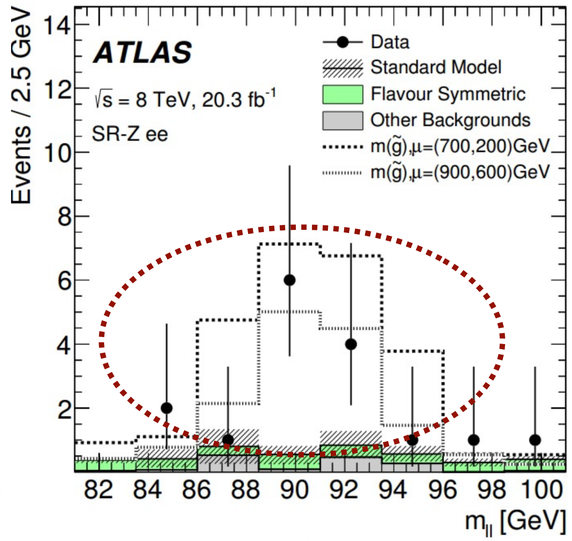
SUSY-2020-15, JHEP 2024

- Reinterpret SUSY analyses using full models.
- Use the Phenomenological Minimal SUSY (pMSSM) [1],[2] with 19 parameters at the high mass scale, imposing theoretical and experimental constraints to reduce the number of parameters.
- Generate a large number of models and explore a large number of final states using existing analyses.
- Point to uncovered regions to which no sensitivity was obtained, design a new round of searches

SMALL EXCESSES

10.1103/PhysRevD.93.052002





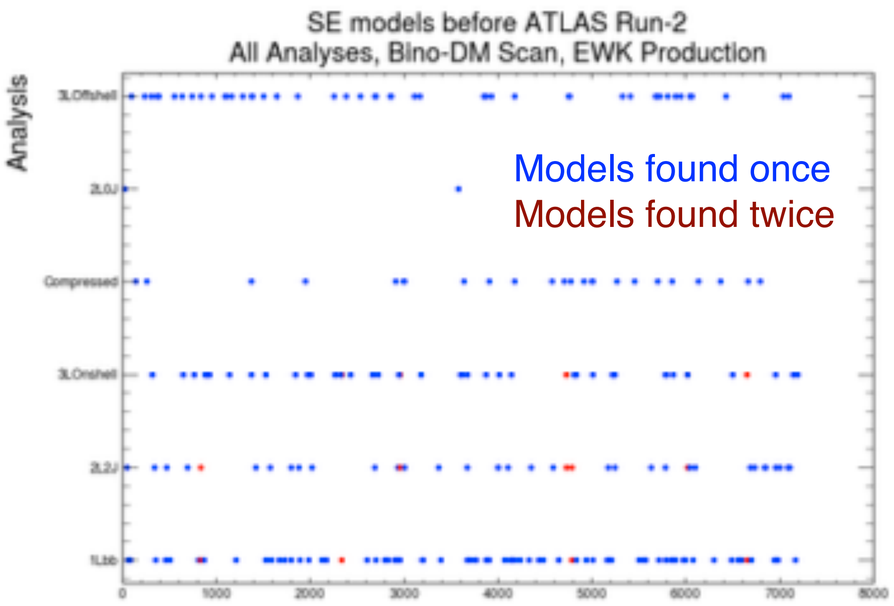
arXiv:1503.03290

Potential new physics:

Good expected sensitivity, but not excluded due to the small excess

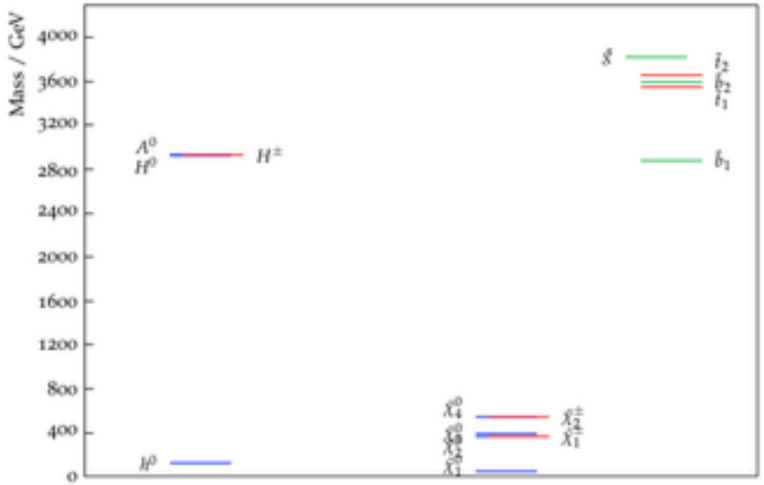
Find a connection for final states and models

ANALYTIC SCAN OF SE MODELS



- ATLAS EWK pMSSM study done in Nishant Gaurav master thesis.
- EWK pMSSM scan (public)
- Single out best motivated models for Run3 searches.

EWK scans:
- SEM 100-200
- Found twice ~10

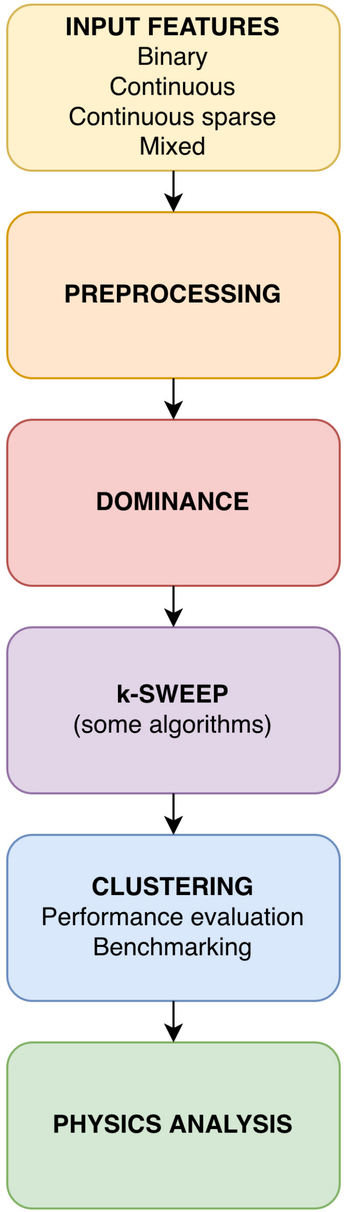


- **New type of study: Look into models that were expected to be excluded, but were not due to small excess (potential new physics).**
- Small Excess analysis criteria: $CL_S^{exp} \leq 0.05$, $CL_S^{obs} > 0.05$, ($Z > 2\sigma$ SD)
- Study properties of the **models** and **final states** used in the analyses that yield a small excess:
- Collectively consider small excesses as hints of new physics
- Exploit connections of small excesses in BSM searches (systematic study)

CLUSTERING OF SE MODELS: WORKFLOW

- Cluster SE models to obtain categories of best-motivated models for future studies
- Unsupervised model categorisation for next round of searches
- Consider highest number of input features (input parameters, masses, BF, cross-section, external constraints, ATLAS analyses survival)
- Optimal clustering algorithm chosen per input type!

Judita Mamužić (IJS) in collaboration with Andrej Filipčić (IJS), Brian Petersen (CERN), Jonas Würzinger (TUM), Lukas Alexander Heinrich (TUM), Nishant Gaurav (Vanderbilt University), Roberto Ruiz De Austri Bazan (IFIC), Aurelio Juste (IFAE), Ciril Bohak (UL FRI), Michele Grossi (CERN)



Examine the input features:
- **Binary**: external constraints and analyses surviving SE criteria
- **Continuous**: mixing, BF
- **Continuous sparse:** mass, input parameters, cross-sections
- **Mixed:** binary+contiuous, binary+sparse contiuous etc.

Determine if the input benefits from **transformation before any clustering** (for any clustering)

Determine if input feature requires a **transformation to mitigate dominance** (important for mixed input features, per clustering algorithm)

Determine the number of clusters (per clustering algorithm, some algorithms have a built-in determination of number of clusters)

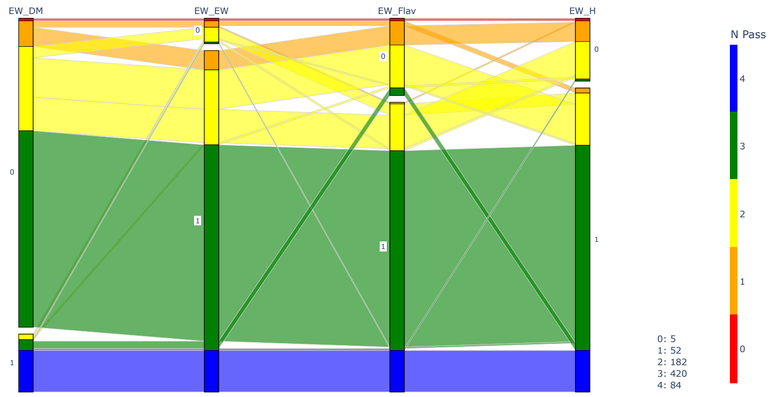
Clustering, **performance evaluation**, **benchmarking** according to typical criteria (per input type and clustering algorithm)

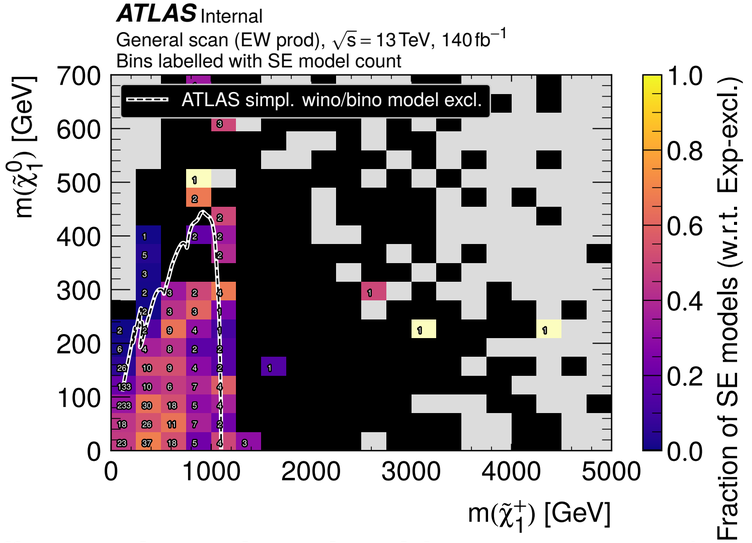
Physics analysis with the optimal clustering algrithm(s)

CLUSTERING OF SE MODELS: PREPROCESSING

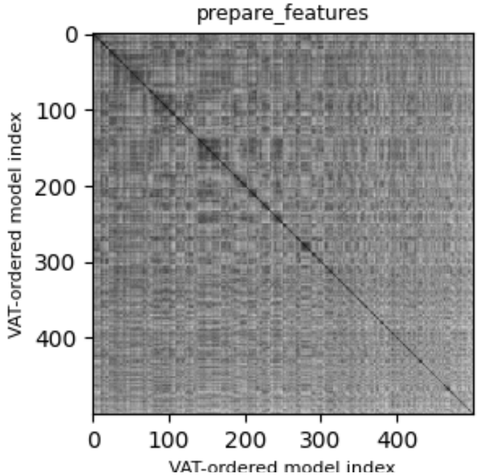
General EW pMSSM scan, SE = True

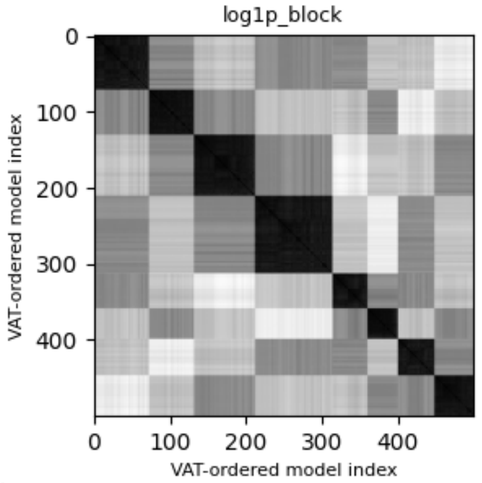
Input



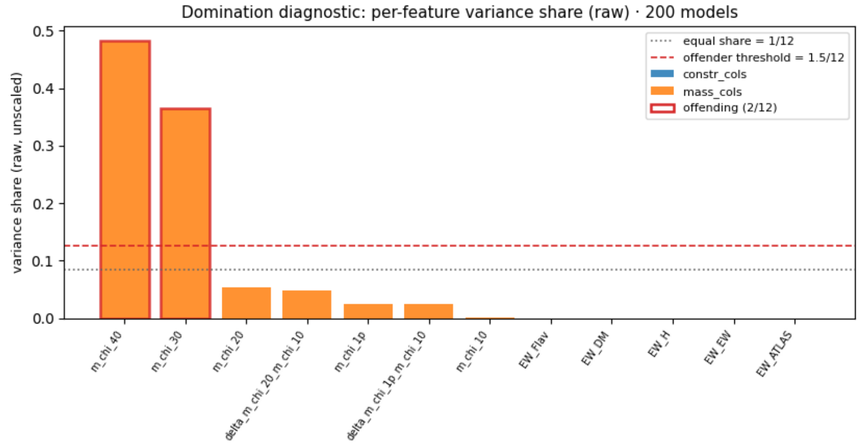


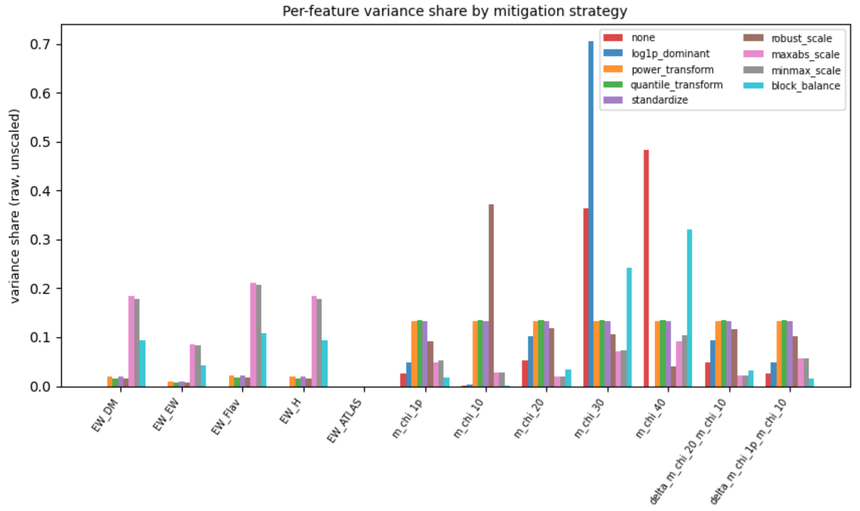
Preprocessing





Domination



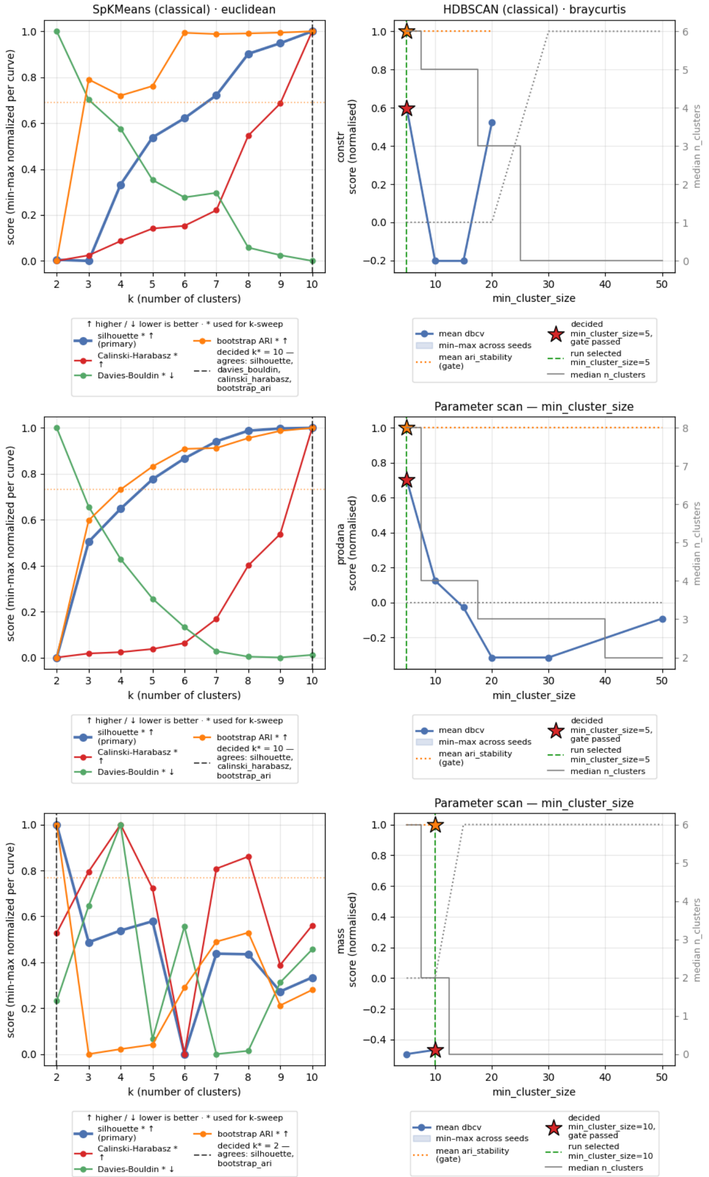


- Different **clustering algorithms** correspond to different inputs:
  - **Binary**: analyses, external constraints
  - **Continuous**: BF, mixing
  - **Continuous sparse**: masses, input parameters, cross section
  - **Mixed**: mixed binary continuous, mixed binary sparse, mixed sparse and continuous, mixed binary all

- Using Log transformation for input parameters
- Using per block and per feature within block normalisation

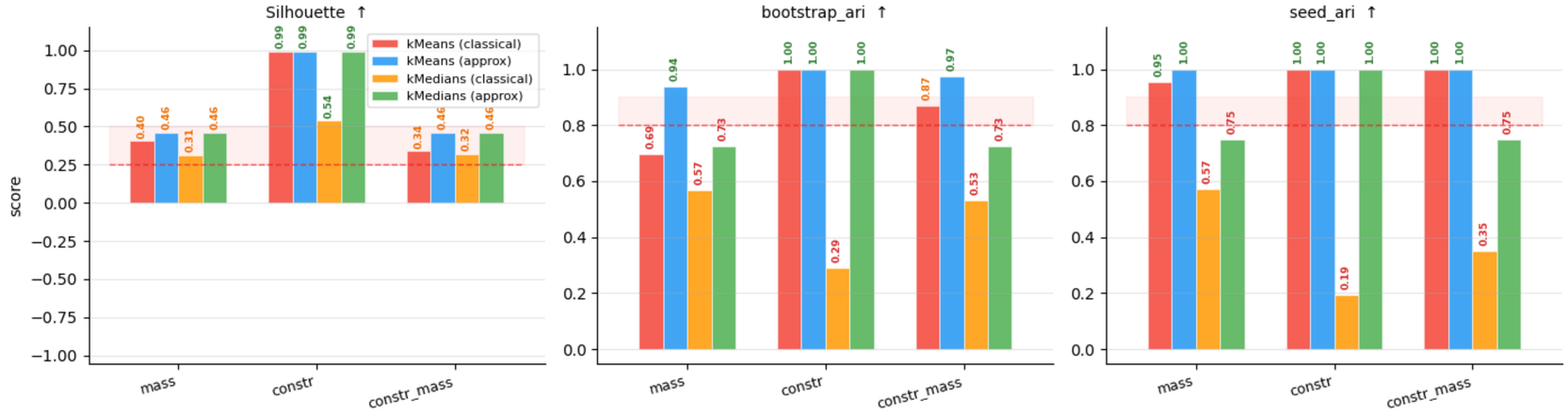
CLUSTERING OF SE MODELS: PERFORMANCE

k-Sweep / builtin

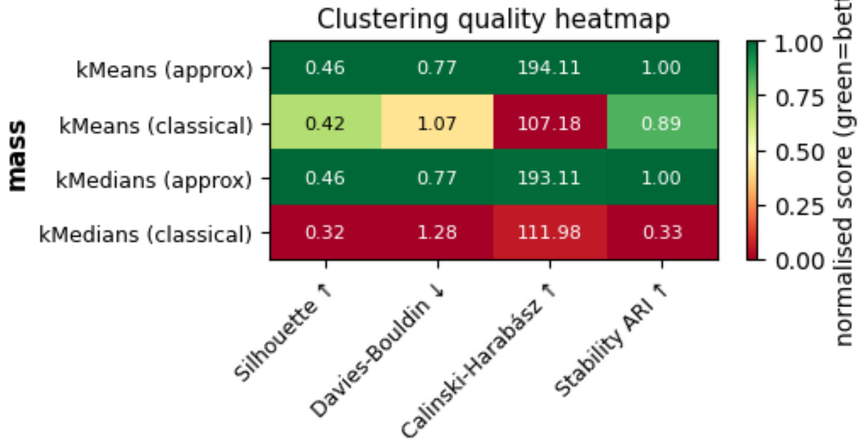


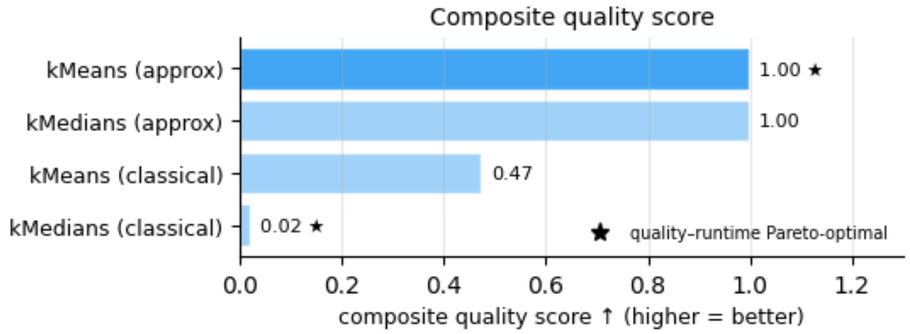
Clustering

Algorithm stability



Algorithm ranking



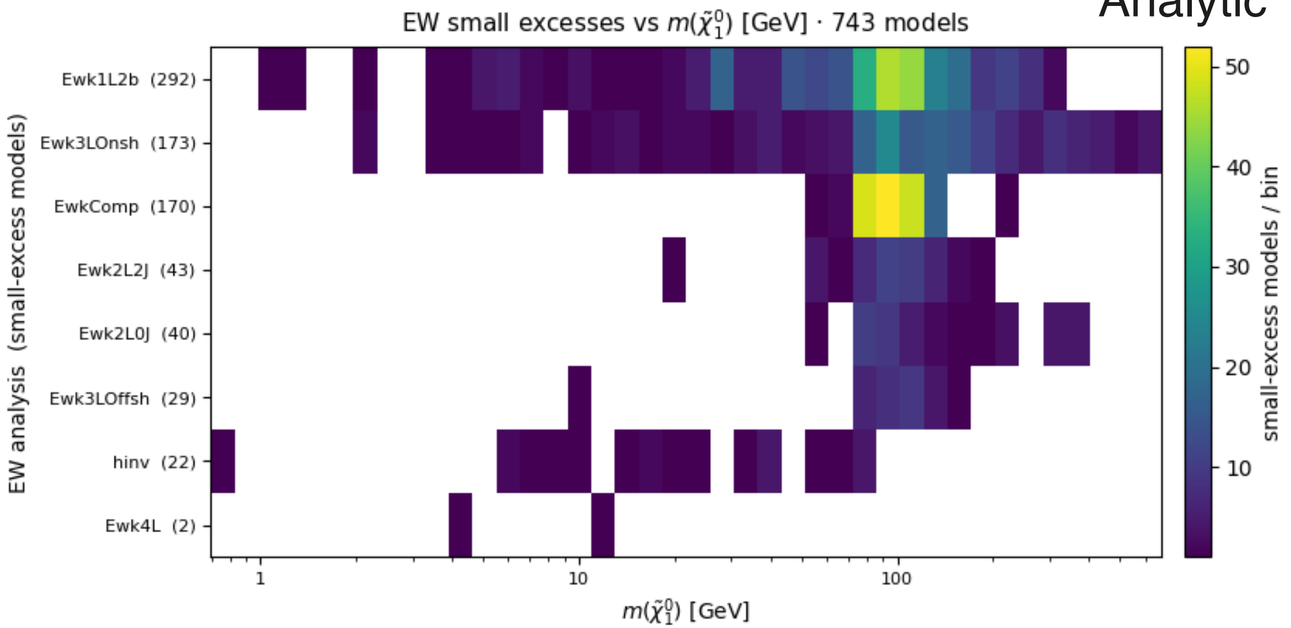


CLUSTERING FOR SE MODELS: APPLICATION

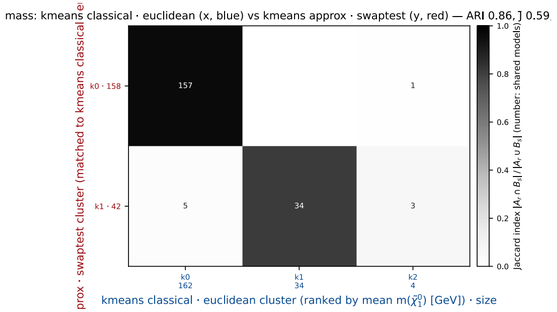
Physics

- Start with a minimal example, gradually extend complexity.
- Choose input blocks, extract physics meaning.
- Analytic:
  - Interesting to see multiple SE models point to ~100 GeV $\tilde{\chi}0_1$
- Clustering, input mass:
  - Dominant cluster has ~100 GeV $\tilde{\chi}0_1, \tilde{\chi}0_2, \tilde{\chi}0_3$ and $\tilde{\chi}\pm_1$ (**considering a higher number of dimensions** and **higher precision in characterisation**)

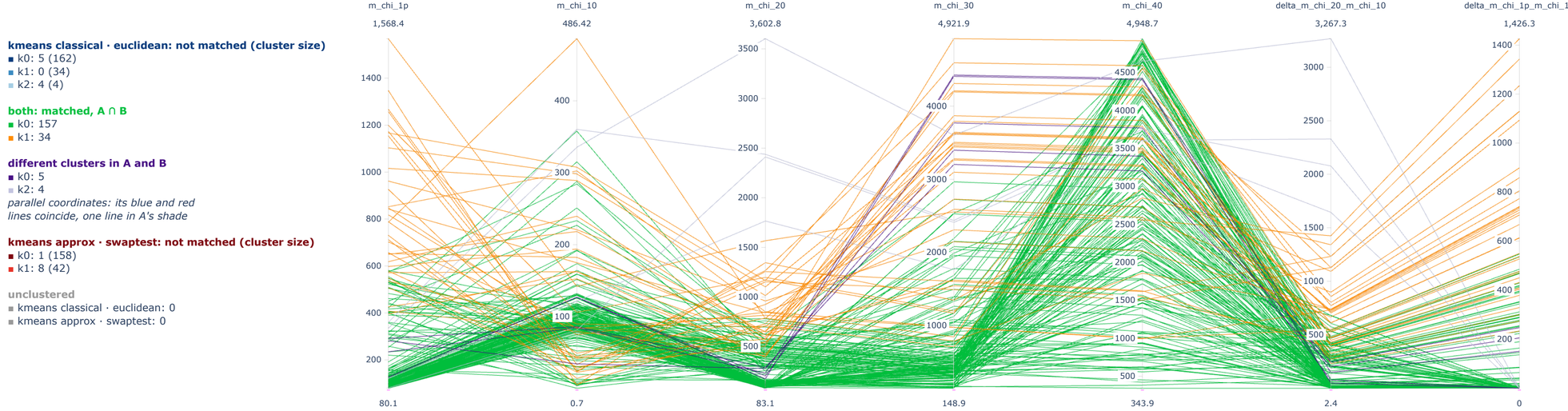
Analytic



Clustering



- Good agreement of two best-performing algorithms

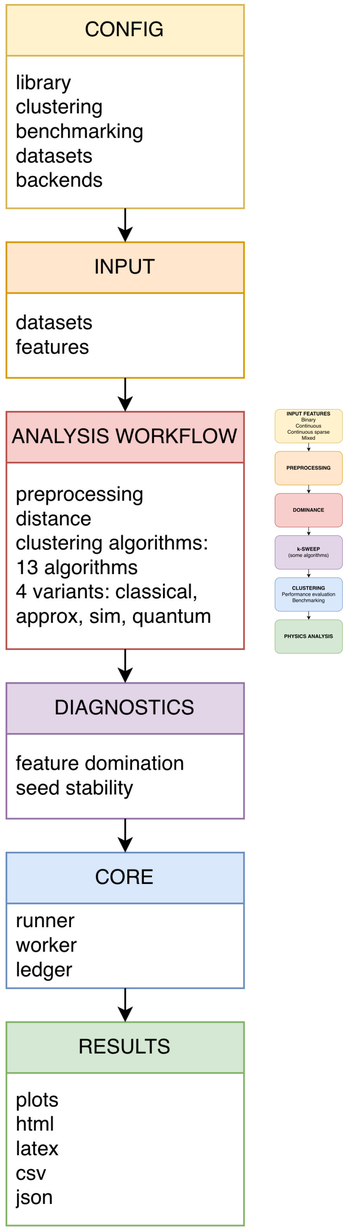


SUMMARY

- **New methodology:**
  - Collectively consider models to which good sensitivity was obtained but were not excluded due to a small excess, potential new physics
- **Clustering of small excess models:**
  - Preprocessing
  - Dominance mitigation
  - Choice of optimal clustering algorithm for given input
  - 2-algorithm cross-validation
- **Physics application**:
  - Example: Dominant cluster points to $\tilde{\chi}0_1, \tilde{\chi}0_2, \tilde{\chi}0_3$ and $\tilde{\chi}\pm_1$ around 100 GeV
- **Next steps:**
  - Summarise methodology and findings (*publication planned in 1 month*)
  - Clustering for data events (model-independent approach, *proof of concept*)
  - Pursuit of new physics using pMSSM (Reinforcement Learning)

BACKUP

QCLUST CODE



src/core/	Handles the core orchestration of the system, including registration, plugins, execution, workers, logging, batching, shared helpers, and the CLI.
src/algorithms/	Contains the common algorithm interface plus family-specific adapters and implementation variants across 13 algorithm families.
src/datasets/ src/features/	Provides dataset-loading utilities and builds the feature groups used by the algorithms.
src/preprocessing.py · distances.py · domination.py	Implements the main dataframe-level preprocessing stages, including distance calculations and domination analysis.
src/diagnostics/ src/output/	Computes diagnostic quality and stability metrics, then stores results and generates reports.
config/	Stores configuration for datasets, execution backends, clustering, and benchmarking.
notebooks/↔run/	Keeps Jupyter notebooks paired one-to-one with executable run scripts, with CI ensuring both versions stay synchronized.
scripts/ scripts/templates/	Contains automation and support scripts for notebook synchronization, CI, flowcharts, shell utilities, batch execution, and benchmark templates.
tests/	Contains the unit test suite that is run by CI on every push.

INPUT

Using preliminary General pMSSM scan, EWK production

CONSTR_COLS:
- EWSummary_ObsCLs_DM_constraints
- EWSummary_ObsCLs_EW_precision_constraints
- EWSummary_ObsCLs_Flavour_constraints
- EWSummary_ObsCLs_ATLAS_Higgs_constraints
- EW_SurvATLAS

PRODANA_COLS:
- Ewk1L2b
- Ewk3LOnsh
- EwkComp
- Ewk2L2J
- Ewk2L0J
- Ewk3LOffsh
- hinv
- EWK4L

XSEC_COLS:
- xsec_20_1p
- xsec_20_1m
- xsec_1p_1m
- xsec_10_1p
- xsec_10_1m

MIX_COLS:
- LSP_Bino_frac
- LSP_Wino_frac
- LSP_Higgsino_frac
- chi_20_Bino_frac
- chi_20_Wino_frac
- chi_20_Higgsino_frac

INPUT_PARAM_COLS:
- M_1
- M_2
- mu
- tanb
- mA
- meL
- meR
- mmuL
- mmuR
- mtauL
- mtauR

MASS_COLS:
- m_chi_1p
- m_chi_10
- m_chi_20
- m_chi_30
- m_chi_40
- delta_m_chi_20_m_chi_10
- delta_m_chi_1p_m_chi_10

BF_COLS:
- BF_chi_1p_to_chi_10
- BF_chi_1p_to_chi_20
- BF_chi_20_to_chi_10_ele_ele
- BF_chi_20_to_chi_10_mu_mu
- BF_chi_20_to_chi_10_Z
- BF_chi_20_to_chi_10_h

CLUSTERING ALGORITHMS

Algorithm	Variant(s)	Implementation	K-selection	K-sweep criteria (validation.ksweep)	Decision for k* (validation.decision)	Primary metric (validation.primary)	Stability tests (validation.stability)
kmeans	classical · approx · sim · quantum	classical: scikit-learn sklearn.cluster.KMeans · approx: in-house NumPy (analytic swap-test distance; classical counterpart of vbelis/latent-ad-qml (https://github.com/vbelis/latent-ad-qml) q-means) · sim: vendored Raijeku/qmeans (https://github.com/Raijeku/qmeans) swap-test/Grover circuits (Qiskit) on qiskit_aer.AerSimulator · quantum: vendored Raijeku/qmeans (https://github.com/Raijeku/qmeans) circuits on IBM Quantum via qiskit_ibm_runtime.SamplerV2	explicit k — sweep / fixed / from_run	scan k (euclidean): silhouette · calinski_harabasz · davies_bouldin · elbow_kneedle · bootstrap_ari	majority vote of silhouette · davies_bouldin · calinski_harabasz · bootstrap_ari; ties → bootstrap_ari	silhouette (max, euclidean)	bootstrap_ari ≥ 0.8 · seed_ari ≥ 0.8
kmedians	classical · approx · sim · quantum	classical: vendored vbelis/latent-ad-qml (https://github.com/vbelis/latent-ad-qml) c_kmedians.Kmedians (modified from GitHub @hounslow) · approx: in-house NumPy (analytic counterpart of kmedians.quantum) · sim: in-house — Raijeku/qmeans (https://github.com/Raijeku/qmeans) swap-test circuits on qiskit_aer.AerSimulator + Weiszfeld geometric median · quantum: vendored vbelis/latent-ad-qml (https://github.com/vbelis/latent-ad-qml) qkmedians (Qibo + TensorFlow)	explicit k — sweep / fixed / from_run	scan k (manhattan): silhouette · calinski_harabasz · davies_bouldin · elbow_kneedle · bootstrap_ari	majority vote of silhouette · elbow_kneedle · bootstrap_ari; ties → bootstrap_ari	silhouette (max, manhattan)	bootstrap_ari ≥ 0.8 · seed_ari ≥ 0.8
spkmeans	classical	cjlin1/sparsekmeans (https://github.com/cjlin1/sparsekmeans) sparsekmeans.LloydKmeans	explicit k — sweep / fixed / from_run	scan k (cosine): silhouette · calinski_harabasz · davies_bouldin · elbow_kneedle · bootstrap_ari	majority vote of silhouette · davies_bouldin · calinski_harabasz · bootstrap_ari; ties → bootstrap_ari	silhouette (max, cosine)	bootstrap_ari ≥ 0.8 · seed_ari ≥ 0.8
spectral	classical	scikit-learn sklearn.cluster.SpectralClustering(affinity="precomputed")	explicit k (n_clusters) — sweep / fixed / from_run	scan k (cosine): eigengap · silhouette_embedding · calinski_harabasz_embedding · davies_bouldin_embedding ·	majority vote of eigengap · silhouette_embedding · bootstrap_ari; ties → bootstrap_ari	silhouette_embedding (max, run distance)	seed_ari ≥ 0.8 · bootstrap_ari ≥ 0.8
hqkmeans	approx	in-house NumPy surrogate of AlessandroPoggiali/QKmeans (https://github.com/AlessandroPoggiali/QKmeans) (HQKMeans-1: binomial shot-noise swap-test distance + Lloyd update)	explicit k — sweep / fixed / from_run	scan k (euclidean): silhouette · calinski_harabasz · davies_bouldin · elbow_kneedle · bootstrap_ari	majority vote of silhouette · davies_bouldin · calinski_harabasz · bootstrap_ari; ties → bootstrap_ari	silhouette (max, euclidean)	bootstrap_ari ≥ 0.8 · seed_ari ≥ 0.8
vqc	approx · sim	approx: in-house NumPy (RBF-kernel k-means; paper-only, no public code) · sim: in-house — qiskit.circuit.library.ZZFeatureMap fidelity kernel on qiskit_aer.AerSimulator	explicit k — sweep / fixed / from_run	scan k (euclidean): silhouette · calinski_harabasz · davies_bouldin · seed_ari	best silhouette, gated by seed_ari ≥ 0.8	silhouette (max, euclidean)	seed_ari ≥ 0.8
vqasc	approx	scikit-learn sklearn.cluster.SpectralClustering on a Gaussian kernel (paper-only, no public code)	explicit k — sweep / fixed / from_run	scan k (euclidean): silhouette · calinski_harabasz · davies_bouldin · seed_ari	best silhouette, gated by seed_ari ≥ 0.8	silhouette (max, euclidean)	seed_ari ≥ 0.8
qclusformer	approx	in-house NumPy random projection + scikit-learn sklearn.cluster.KMeans (paper-only placeholder, no public code)	explicit k — sweep / fixed / from_run	scan k (euclidean): silhouette · calinski_harabasz · davies_bouldin · seed_ari	majority vote of silhouette · davies_bouldin · calinski_harabasz · seed_ari; ties → seed_ari	silhouette (max, euclidean)	seed_ari ≥ 0.8
qubosa	approx	in-house NumPy QUBO k-medoids + simulated annealing (transcription of Bauckhage et al. 2018, Eqs. 1–4)	explicit k — sweep / fixed / from_run	scan k (euclidean): silhouette · calinski_harabasz · davies_bouldin · elbow_kneedle · seed_ari	majority vote of silhouette · davies_bouldin · calinski_harabasz, gated by seed_ari ≥ 0.8; ties → seed_ari	silhouette (max, euclidean)	seed_ari ≥ 0.8
hdbscan	classical	scikit-learn sklearn.cluster.HDBSCAN(metric="precomputed")	implicit — auto: k emerges from the density tree (min_cluster_size, min_samples)	scan min_cluster_size (grid [5, 10, 15, 20, 30, 50]): dbcv · n_clusters · noise_frac	best dbcv, gated by ari_stability ≥ 0.8	dbcv (max, run distance)	bootstrap_ari ≥ 0.8
leiden	classical	vtraag/leidenalg (https://github.com/vtraag/leidenalg) leidenalg.find_partition + CPMVertexPartition on a python-igraph kNN graph	implicit — auto: k = number of communities (k_neighbors graph, resolution)	scan resolution (grid [0.1, 0.25, 0.5, 0.75, 1.0, 1.5, 2.0]): modularity · n_clusters	best modularity, gated by ari_stability ≥ 0.8	modularity (max)	seed_ari ≥ 0.8
qclue	approx	in-house NumPy surrogate (algorithms._density.density_fit) of QLUE-algo/qlue (https://github.com/QLUE-algo/qlue)	implicit — auto: k = density critical points (dc, rho_c, delta)	scan dc (grid from distance quantiles): dbcv · n_clusters · noise_frac	best dbcv, gated by ari_stability ≥ 0.8	dbcv (max, run distance)	seed_ari ≥ 0.8
qlue	approx	in-house NumPy surrogate (algorithms._density.density_fit) of QLUE-algo/qlue (https://github.com/QLUE-algo/qlue)	implicit — auto: k = density critical points (dc, rho_c, delta)	scan dc (grid from distance quantiles): dbcv · n_clusters · noise_frac	best dbcv, gated by ari_stability ≥ 0.8	dbcv (max, run distance)	seed_ari ≥ 0.8

PREPROCESSING

Input type	Test	Comment
Binary	**Number of distinct patterns**	N models per distinct flag pattern — discrete structure a binary clusterer can actually find: **few heavy patterns = natural candidate clusters** many near-singleton patterns = little to cluster
(Sparse) Continuous	**Hopkins statistics**	**≈ 0.5 no structure, → 1 clusterable, → 0 regular grid** (Hopkins & Skellam 1954, **Ann. Bot.** 18:213; Banerjee & Davé 2004, FUZZ-IEEE)
	Pairwise distance and cofficient of variation	Measures how useful pairwise distances are for clustering using relative contrast and coeffiecient of variation, **very large values indicate likely near-duplicate rows**.
	VAT ordered disssimilarity	Reorders the dissimilarity matrix so similar observations appear next to each other, making cluster structure visible. **Clear dark diagonal blocks suggest distinct clusters**; **sharper blocks after preprocessing indicate improved** separation, while **diffuse patterns suggest weak structure**.

Transformations:

- **Log** — compresses large values and reduces skew.
- **PCA** — projects data onto directions of maximum variance.
- **UMAP** — creates a low-dimensional embedding that preserves local structure.
- **t-SNE** — creates a low-dimensional embedding emphasizing local neighborhood separation.

DOMINANCE

Mitigation	Comment
Log	Compresses large values with log transformation..
Robust scale	Rescales using the median and interquartile range.
Block balance	Rescales each dominant feature block to unit total variance while preserving relative variance within the block.
Max-absolute scale	Normalizes each feature by its maximum absolute value.
Min-max scale	Maps each feature to a fixed bounded range.
PCA Whiten	Logarithmically compresses only the most dominant feature(s).
Quantile transform	Maps features to a common target distribution.
Standardize	Rescales features to zero mean and unit variance.
Power transform	Gaussianizes feature distributions and stabilizes variance.

CLUSTERING K-SWEEP AND PERFORMANCE TESTS

Criterion	Implementation
silhouette / silhouette_embedding	sklearn.metrics.silhouette_score (embedding variant scored in sklearn.manifold.spectral_embedding)
calinski_harabasz (_embedding)	sklearn.metrics.calinski_harabasz_score
davies_bouldin (_embedding)	sklearn.metrics.davies_bouldin_score
elbow_kneedle	kneed.KneeLocator on the inertia curve
bootstrap_ari · seed_ari · ari_stability	sklearn.metrics.adjusted_rand_score (mean pairwise over refits)
eigengap	scipy.linalg.eigh + sklearn.manifold.spectral_embedding
dbcv	hdbscan.validity.validity_index
modularity	igraph.Graph.modularity
n_clusters · noise_frac	NumPy label counts (diagnostics/ksweep.py::compute_param_scan)

Algorithm stability:

- **Silhouette score** — measures **how well each sample fits within its assigned cluster** relative to neighboring clusters, with higher values indicating better separation and cohesion
- **Bootstrap ARI** — measures **clustering stability across resampled datasets** using the Adjusted Rand Index, with higher values indicating robust cluster assignments
- **Seed ARI** — measures **clustering stability across different random initializations** using the Adjusted Rand Index, with higher values indicating low sensitivity to the random seed

Clustering performance:

- **Silhouette score** — measures **within-cluster cohesion relative to separation** from neighboring clusters; **higher is better**
- **Davies–Bouldin index** — measures **average similarity between each cluster** and its most similar competing cluster; **lower is better**
- **Calinski–Harabasz score** — compares between-cluster **dispersion** with within-cluster dispersion; **higher is better**