Supplementary Figure 1

Features for predicting protein interaction interfaces.

(a) A schematic showing the five feature categories from which feature sets are optimized to train ECLAIR. (b) The portions of high-quality binary interactomes for which each feature type is available. (c) Feature aggregation strategies employed for combining multiple points of evidence into single co-evolution- and structure-based features. For co-evolution, we select the top co-evolved residue, the mean of features for the top 10 co-evolved residues, or the mean over all co-evolved residues in the partner protein. For proteins with multiple structures, we take the mean, minimum, or maximum SASA over all available structures.
Supplementary Figure 2

Testing set-training set feature balance.

Balance between testing/training and prediction sets of sequence- and structure-based feature depths. (a) Sources (PDB or ModBase) and number of structures used to calculate solvent-accessible surface area. (b) Number of homologous sequences used to calculate evolutionary features. (c) Sources of docked models for calculating docking-based features.
Supplementary Figure 3

A comparison of two methods for handling missing data in classification.

A comparison of (1) imputation and (2) an ensemble of fully-trained classifiers for handling missing data. During training, imputation must fill in gaps in feature coverage, whereas an ensemble trains independent classifiers on each feature-availability scenario. Since structural feature coverage is highly correlated with the existence of known interface residues in training, imputation will fail to predict interface residues outside of regions with structural feature coverage (red). An ensemble will predict interface residues based only on the features available and will not be biased by the missing structural feature.
Supplementary Figure 4

Training and optimizing the ECLAIR classifier.

(a) Training the ECLAIR classifier. (b) Four methods for optimizing machine learning algorithm hyperparameters, showing the order of trials and granularity of hyperparameter sampling spaces for optimizing two hyperparameters. (c) Cross-validation strategy using TPE to optimize hyperparameters and window sizes for both feature selection and ensemble classifier training. (d) Cross-validation results using TPE trials to select top performing feature or set of features (in red) in each feature category. (e) Comparison of four hyperparameter optimization methods’ performance (top panel) and hyperparameter and residue window sampling patterns (bottom panels) on one of the eight sub-classifiers of the ECLAIR ensemble.

Nature Methods: doi:10.1038/nmeth.4540
Supplementary Figure 5

ECLAIR predictions.

(a) Number of residues predicted in each prediction confidence category. (b) Cumulative distribution of interactions with ≥ n residues classified as interface for each of the highest interface potential categories.
Supplementary Figure 6

Performance of ECLAIR sub-classifiers on testing set.

(a) Receiver operating characteristic (ROC) curves for each sub-classifier. (b) Precision-recall curves for each sub-classifier. (c) Distribution of raw prediction scores for each sub-classifier. For all panels, sub-classifiers plotted in blue used only sequence-based features; sub-classifiers in red used additional structure-based features. (d) Raw prediction scores compared to actual probabilities of residues in each bin to be at the interface.
Supplementary Figure 7

ROC and precision-recall curves comparing ECLAIR with other popular interface residue prediction methods.

Here, only known surface residues were used in benchmarking all methods. All methods have a slightly lower AUROC (since it is more difficult to distinguish interface from non-interface among only surface residues), however ECLAIR still performs as well or better than all tested methods.
Supplementary Figure 8

Genomic properties of predicted interface residues in interactions lacking structural features.

(a) Enrichment of disease mutations in predicted and known interfaces. (b) Enrichment of recurrent cancer mutations in predicted and known interfaces. (c) Enrichment of rare and common population variants in predicted and known interfaces. (d) Predicted deleteriousness of population variants in known and predicted interfaces (using PolyPhen-2). (e) Predicted effects of population variants in known and predicted interfaces (using EVmutation). (In a-b, significance determined by two-sided Z-test. In d-e, significance determined by a two-sided U-test. n.s. denotes not significant)
Supplementary Figure 9

Genomic properties of predicted interface domains.

(a) Enrichment of disease mutations in predicted and known interfaces. (b) Enrichment of recurrent cancer mutations in predicted and known interfaces. (c) Enrichment of rare and common population variants in the predicted and known interfaces. (d) Predicted deleteriousness of population variants in known and predicted interfaces (using PolyPhen-2). (e) Predicted effects of population variants in known and predicted interfaces (using EVmutation). (In a-b, significance determined by two-sided Z-test. In d-e, significance determined by a two-sided U-test)
Supplementary Figure 10

Other properties of predicted interface domains.

(a-c) Precision recall curves for interfaces predicted with ECLAIR: (a) interface residues in all benchmarked interactions, (b) interface residues in interactions lacking structural features, and (c) interface domains in interactions lacking structural features. (d) Fraction of interface residues localized to domains for known interface residues in co-crystalized co-bound proteins, predicted interface residues in interactions with structural features, and predicted interface residues in interactions without structural features. (e) Enrichment of human disease mutations in domains determined by known interface residues in co-crystalized co-bound proteins, predicted interface residues in interactions with structural features, and predicted interface residues in interactions without structural features. (Significance determined by two-sided Z-test)
Supplementary Figure 11

Uncropped gel image from Figure 5a.
Supplementary Note 1: Curating interactomes

Interactomes today can be curated from several sources. For instance, some databases scour the literature and receive direct submissions from both large-scale, high-throughput studies (using assays such as Yeast two-hybrid\(^1\) and affinity purification/mass spectrometry\(^2\)), and small, targeted studies (using assays such as Co-IP\(^3\)). Both sources are very valuable to systems biology. Large-scale screens, for instance from the CCSB consortium\(^4\), attempt to discover whole or large portions of interactomes all at once, by testing all potential pairs of proteins for interactions. These results are unique because they produce sets of interactions that are less susceptible to study bias than chosen small-scale studies and provide the first glimpses of network architecture from which we can elucidate principles of evolution and disease\(^5\). Small-scale studies help fill gaps in interactomes and reveal specific biological activity in pathways in a hypothesis-driven manner. Databases then make available these interactions, indicating which type of study each is derived from using experimental evidence codes.

In curating interactomes for this study, we have collected all experimentally-determined protein interactions from 7 primary interaction databases, including IMEx\(^6\) partners DIP\(^7\), IntAct\(^8\), and MINT\(^9\), IMEx observer BioGRID\(^10\), and additional sources iRefWeb\(^11\), HPRD\(^12\), and MIPS\(^13\). This selection of databases is representative of all binary interactions available through the IMEx consortium and PSICQUIC web portal\(^14\), either through direct integration of each databases’ curated data, or inclusion of databases that contain interactions from other databases. We distinguish between binary interactions and co-complex associations by interpreting the PSI-MI evidence codes associated with each reported interaction (See Supplementary Table 2 for a complete list of binary evidence codes). Because it cannot be determined which two proteins share a physical binding interface for co-complex associations, we only predict interfaces for binary interactions. We typically only include interactions that have been reported by two or more independent publications, as these interactions have been shown to be more reliably reproducible\(^15\). However, to spur genomic studies in the single most complete set of experimentally determined binary interactions, we have not performed this filtering for the human interactome, and have retained all 122,647 binary interactions from all source databases.

Nature Methods: doi:10.1038/nmeth.4540
Supplementary Note 2: Current methods for interaction interface prediction

2.1 Current methods, strategies, and features for interface prediction

The reason no method so far has been able to predict interaction interfaces for entire interactomes is that most studies focus on exploring the efficacy of feature combinations and classification strategies\textsuperscript{16-18}, rather than predicting for large portions of the interactome to fuel genomic studies. Still, this has produced crucial knowledge about predicting interaction interfaces for subsets of interactomes that meet certain feature requirements. While biophysical sequence properties are available for any interaction and have been used with great regularity in predicting interfaces, many other features used for protein interaction interface prediction are not available for all interactions. For instance, protein structure-based predictors, such as PredUS\textsuperscript{19}, PINUP\textsuperscript{20}, PIER\textsuperscript{21} and others\textsuperscript{18} have explored the use of known protein surface, homology, and surface predictions to inform interface prediction, and CPORT\textsuperscript{22} and RCF\textsuperscript{23} have enabled the incorporation of molecular docking. Other studies have shown the value of co-evolutionary analyses for predicting intra- and inter-protein contacts\textsuperscript{24,25} and combinations of docking with co-evolutionary features have been successfully demonstrated on a small scale by SCOTCH\textsuperscript{26} and in conjunction with RosettaDock\textsuperscript{27}.

Over the past few years, some methods have explored new variations on algorithmic approaches, including revisiting the use of deep learning\textsuperscript{28}, and the use of a novel ensemble of SVM and random forest algorithms\textsuperscript{29}. Innovative applications of features also continue to be an active area of exploration: Várnai \textit{et al.} show that even multiple sequence alignments (MSAs) of relatively limited depth can be used to produce informative features\textsuperscript{30}, and Garcia-Garcia \textit{et al.} demonstrate a novel homologous sequence fragment approach that does not require deep MSAs\textsuperscript{31}. There are several recent, comprehensive reviews of some of these and many other machine learning methods for predicting protein interaction interface residues\textsuperscript{18,32,33}. Other methods focus on predicting entire 3D structures of protein-protein interactions or complexes through template-based approaches\textsuperscript{34,35}, molecular docking\textsuperscript{22,36-38}, and combinations of the two\textsuperscript{39-41}. Such modeling approaches have been benchmarked in the CAPRI competition\textsuperscript{42}, and their scoring methods are surveyed in a recent review\textsuperscript{43}. Finally, methodological advances in feature construction and algorithm training have been applied to problems beyond intra-species binary protein-protein interface prediction. For instance, several studies have explored using similar frameworks for predicting host-pathogen interactions\textsuperscript{44,45}, and others have been used to predict contacts of large protein complexes\textsuperscript{25,46}.

While these and many other conceptual advances have furthered the field of protein interaction interface prediction, the focus thus far has been on refining the use of features for small portions of the interactome, rather than producing predictions for whole interactomes. We aim to combine the knowledge gained from past studies to provide a single, unified framework for predicting interfaces in complete
interactomes that will be amenable to functional genomic analyses using our new web tool, Interactome INSIDER.

2.2 Genomic validation of interaction interfaces

Validation of interface prediction methods typically entails the prediction of interfaces in a set of interactions for which interfaces are known but have not been supplied to the algorithm for purposes of training. Often these interactions used for validation are chosen to be a community benchmark set, such as the docking benchmark set used in this study\textsuperscript{47}, which enables comparison of many methods on a more even playing field. Though these community benchmarks are used ubiquitously, it is still important to consider whether all methods can be applied without creating a handicap or an improper advantage for some compared to others. For instance, co-evolutionary features used in ECLAIR can only be correctly applied to naturally occurring, intra-species protein interactions, which will hinder its performance on benchmarks that contain inter-species or synthetic interactions.

Biological validation of interaction interface predictions, wherein functional genomic properties of predicted interfaces (i.e. localization of disease and cancer mutations or population variants) are measured or tested in-vitro, is less commonly performed, but is critical for understanding whether predictions can be used in genomic studies. To our knowledge, no method for predicting interface residues has been validated using genomic properties, as we have performed with ECLAIR. However, both biophysical\textsuperscript{48-51} and genomic\textsuperscript{52-54} properties of residues at known interfaces have been thoroughly explored, establishing a baseline understanding of true interfaces to which predicted interfaces can be compared.

There are many studies that predict genomic properties of variants, for instance, deleteriousness predictors such as Polyphen-2\textsuperscript{55}, SIFT\textsuperscript{56}, Mutation Taster\textsuperscript{57}, and Mutation Assessor\textsuperscript{58}. Other methods predict specific effects of mutations by modeling their disruption to interaction domains and interfaces. For instance, BeAtMuSiC\textsuperscript{59}, SAMMBE\textsuperscript{60}, and BindProfX\textsuperscript{61} model the effects of mutations on protein interactions primarily by predicting free energy change resulting from mutations. Modeling of free energy, domain architecture\textsuperscript{62}, and variant deleteriousness has already vastly extended the reach of genomic studies to include proteins and interactions that would be expensive or impossible to study through more direct experimental techniques\textsuperscript{35,52,63,64}. With Interactome INSIDER, we aim to provide the same level of coverage for predicted interfaces, so that scientists can combine the knowledge from all of these modeling efforts to inform genomic studies.

Nature Methods: doi:10.1038/nmeth.4540
**Supplementary Note 3: Feature selection and engineering**

In order to build a machine learning classifier to predict whether or not an amino acid residue is at the interface of an interaction, we selected an initial set of features widely reported\(^\text{18}\) to be predictive of interface residues: (1) ensemble biophysical features, such as residue hydrophobicity and polarity, (2) evolutionary sequence conservation, and (3) solvent-accessible surface area (SASA) (Supplementary Fig. 1a). Of these features, SASA is widely reported to be most informative, since all interface residues, by definition, are on the surface of proteins. However, methods using SASA typically take experimentally-determined structures as inputs, severely limiting their applicability to full-length proteins and those without structural coverage in the PDB. To combat this issue and vastly increase structural coverage, we also compute SASA from single protein homology models\(^\text{34}\).

One key missing aspect from this set of features and from many interface predictors is the incorporation of interaction partner-specific information. Methods that rely on just the aforementioned features will predict the same interface residues for a protein, regardless of its binding partner. To our knowledge, there are no current interface prediction methods that combine partner-specific features with other protein features to predict on a large scale, but several recent studies have described suitable methods for calculation of partner-specific features\(^\text{23,25}\). To build ECLAIR, we incorporate two classes of partner-specific features: (1) co-evolution of amino acid sequences and (2) computational molecular docking (Supplementary Fig. 1a).

The basis for use of evolutionary sequence-based features is the postulate that highly conserved sites in orthologous proteins are likely to be functionally important, and since much of protein function is determined by interactions with other proteins, conserved residues are more likely to be interface residues\(^\text{65}\). Co-evolutionary features summarize dependent patterns of conservation in two interacting proteins. In addition to being well-conserved, we expect that residues critical to maintaining the interaction interface should evolve in a coordinated manner so as to maintain binding complementarity. Proven methods of measuring co-evolution include statistical coupling analysis (SCA)\(^\text{66}\) and direct coupling analysis (DCA)\(^\text{67}\), both of which are used to produce features to train our classifiers.

Molecular docking is a class of stand-alone method to predict bound conformations of protein interactions by computationally sampling low-energy binding conformations of individual structures of interacting proteins. Docking is highly specialized and computationally intensive, making it traditionally ill-suited to interactome-scale prediction on its own. However, we incorporate docking results into our classifier by synthesizing features that attempt to capture a summary of the low-energy orientations of structural subunits in relation to one another, which we sampled by a docking program.

The ECLAIR classifier is built using all five of the aforementioned feature categories: (1) biophysical properties, (2) conservation, (3) co-evolution, (4) SASA, and (5) docking (Supplementary Fig. 1b).
From each of these categories, we synthesized many variations of features using feature engineering techniques such as scaling, aggregation, and combination. Scaling refers to the use of either raw calculated values (i.e. the absolute SASA in Å²) or normalized values (i.e. the Jensen-Shannon divergence of each residue normalized to the average of all positions per protein). Aggregation strategies dictate how features derived from multiple sources of evidence (i.e. multiple protein structures for computing SASA) are combined to form a single feature (Supplementary Fig. 1c). Finally, multiple types of aggregation and/or scaling strategies can be used to produce several feature variations of the same raw feature to train a classifier if they are shown to have an additive effect on predictive performance.

3.1 Feature Calculation

Biophysical Residue Properties: Seven per-residue biophysical sequence features are derived from ExPASy ProtScale: (1) hydrophobicity, (2) polarity, (3) average buried area from protein folding, (4) molar fraction of accessible residues, (5) transmembrane tendency, (6) bulkiness, and (7) amino acid composition. Each residue in interacting proteins is assigned the value for that residue based on the published scales.

Evolutionary sequence conservation: We use PSI-BLAST to find homologs of each input protein across all proteins in the UniProt database. We retain the highest-scoring homolog by e-value for each organism in the PSI-BLAST output, with an e-value cutoff of 0.05. We then produce a multiple sequence alignment (MSA) between the original sequence and the retained UniProt sequences using Clustal Omega. For each protein with at least 50 homologs, we calculate the Jensen-Shannon divergence across all positions in the original protein sequence.

Co-evolution: Using the MSAs produced for the calculation of sequence conservation, we perform a co-evolutionary analysis of compensatory residue changes in interacting proteins. For each interaction, we search the MSAs of both proteins for homologs from the same organism. For interactions that have at least 50 homologs from the same species for both interacting proteins, we concatenate the homologous sequences end-to-end. We then produce a new MSA containing concatenated sequences from each species for which there were homologs for both proteins. Using this MSA, we perform DCA and SCA to evaluate the extent to which sequence positions between proteins are correlated in the context of evolution. For DCA, we calculate both the direct information and mutual information scores. Because it is impossible to disentangle intra- from inter-protein co-evolution for homodimers, we only perform co-evolution analyses for heterodimers. While the cutoff of 50 homologous sequences is lower than those used in several studies, we have shown that such a cutoff is appropriate, as it enables our ECLAIR classifier to use binding partner-specific features for many more interactions. Additionally, we have shown the capability of these features to distinguish interfaces of a protein with multiple binding
interfaces (Fig. 5-6). Furthermore, we performed a study of the effect that the number of homologs used for co-evolutionary calculations has on the ultimate performance of our classifier, and conclude that the addition of co-evolutionary features is a boon to classifier performance for interactions with both low and high numbers of homologs (Supplementary Table 6).

**Surface Residues:** We curate both experimentally determined crystal structures from the PDB \(^{72}\) and homology models from ModBase \(^{34}\) for each protein. For PDB structures, we calculate the solvent accessible surface area (SASA) using NACCESS \(^{73}\) of all PDB chains which contain UniProt sequences, each chain in isolation. For ModBase models, we calculate SASA for all models with ModPipe Quality Score (MPQS) ≥ 1.1. For each protein, we average the raw values of SASA for each UniProt position covered by either a PDB structure or a ModBase model into a single feature.

**Molecular Docking:** Using the molecular subunits identified for the calculation of SASA from both the PDB and ModBase, we performed rigid-body molecular docking using zDOCK \(^{36}\). We performed docking for interactions where both proteins have a PDB structure covering ≥50% and 50 residues of each UniProt sequence or a ModBase model with the same sequence coverage and MPQS ≥1.1. In each case, docking was performed on the pair of structures with the highest overall UniProt residue coverage using zDOCK to produce 2,000 conformations of the subunits in complex. Docking results are encoded as features by calculating the distance between each residue and the closest residue of the other subunit across the top 10 docking results. In total, we performed docking for 29,896 interactions in our prediction set and 4,424 in our training and testing sets using >50,000 core-hours of distributed computing on Amazon Elastic Compute Cloud (ec2) instances.

### 3.2 Feature Engineering

Several strategies of transformation and aggregation of features were tested during cross-validation to determine which combinations were most predictive of interface residues.

**Aggregation:** Features that were collected from multiple sources were aggregated using a min, max, mean, or top strategy. SASA was collected across all available models, with each residue feature defined as either the maximum or mean SASA observed at that residue across all models. Docking features were aggregated as either the minimum, maximum, or mean distance from each residue to the opposing protein across the top 10 docked models. Co-evolution features were encoded for each residue as either the maximum, mean, or mean of the top 10 of the co-evolution scores with all residues in the interacting protein.

**Normalization:** Both raw feature values and normalized feature values were tested for many feature categories. Normalization was performed by calculating the z-score (number of standard deviations from
the mean) of a residue feature in relation to all other such residue features in a given protein. When aggregation and normalization were performed together, aggregation preceded normalization.

3.3 Coarse-grained feature use

Two of the feature types we use to predict interfaces, docking and co-evolution, are specialized and highly sensitive to input parameters. This is likely the reason neither has been used on its own for prediction of interfaces on a large scale, as it is simply too time-intensive and error-prone to use them in an unattended manner for prediction for any interaction. For instance, docking may require human intervention for some cases\textsuperscript{27}, and co-evolutionary analyses are often bound by the number of available homologous sequences\textsuperscript{24}. However, for machine learning, we have found that a broad large-scale application of docking and co-evolution as coarse-grained features can still produce informative results when combined with other features. These features are thus not used directly for predicting interfaces (as they have been by other methods\textsuperscript{24,36}), but as a more subtle influence to a final interface prediction. Given that docking and co-evolution are the only features in ECLAIR that have the potential to distinguish interfaces of a protein with multiple binding partners, it is important that we incorporate them in this way. We have shown this strategy to be successful, given these features’ ability to produce differential interface predictions that mirror the functions of differential interfaces in experimentally determined structures (Fig. 5c-d). In the case of molecular docking, it has been shown that an automated combination of docking poses can be used to predict protein interfaces\textsuperscript{23}, and while co-evolution metrics are often calculated for very deep multiple sequence alignments\textsuperscript{25,67}, we have shown their contribution to the performance of our classifier for both high and low homology depths (Supplementary Table 6).

3.4 Feature Balancing

Features derived from multiple independent points of evidence include SASA and docking, each of which may be determined from multiple models and from models derived from either the PDB, ModBase, or both. To ensure fair evaluation of the classifier, we checked and rebalanced the composition of these features by controlling the quantity and source of models used to compute each feature in the training and testing sets to best approximate the composition of the prediction set, without leaving out any interactions. We also determined there to not be significantly fewer sequences used in alignments of interactions in the prediction set compared to the testing and training sets (Supplementary Fig. 2).
Supplementary Note 4: Constructing the ECLAIR ensemble classifier

Because not all features are available for all residues, a single classifier trained on all of the aforementioned features would have limited applicability to full interactomes. For instance, current methods trained using structural features will be unable to predict for residues without structures, even when other features are available. Imputation methods can be used to fill in missing features, allowing a single machine learning algorithm to be applied to all data (Supplementary Note 5). For instance, imputation methods have been used to resolve gaps in micro-array data. However, these methods rely on two assumptions: (1) relatively high feature coverage, and (2) missing features are distributed randomly, without respect to the label of the data in the training set (i.e. whether the residue is at the interface). The first assumption is clearly violated for predicting interface residues, as >50% of some structural features are missing in interactomes (Supplementary Fig. 1b).

Perhaps more importantly, features are not missing at random. Due to the technical challenges of crystallizing proteins, some proteins and protein regions are inherently more likely to be available in the PDB. Those protein regions that are available in co-crystal structures, which are used to assess true interface residues for training the algorithm, are also more likely to be available in other crystal structures used to gather structure-based features for predicting. Therefore, there will be very few examples of true interface residues in regions without structural features. A classifier trained using imputation to fill-in missing SASA features would then be able to use the fact that the SASA feature is missing to always predict that the residue is not at the interface. During training, this will increase the apparent performance of the algorithm due to the classifier having learned this pattern in the training data where single crystal structures are likely to mirror the coverage of co-crystal structures; but during prediction for interactions with unknown interfaces, the algorithm will be unlikely to predict any interface residue outside of a region or full protein that has not been crystalized, which will lead to increased misclassification (Supplementary Fig. 3).

To address this issue, we trained an ensemble of 8 independent classifiers built on 8 subsets of features covering likely cases of feature availability. We then only use a single classifier to predict for each residue—the classifier that was trained (without imputation) using other residues with the same set of features (Supplementary Fig. 3). When predicting for an interaction missing structure-based features, a classifier trained only using sequence-based features will be used, thereby allowing predictions of interface residues based only on the available features and not on the lack of structural features. In this way, the bias introduced by imputation is eliminated since missing values simply require that a different classifier in the ensemble is used.
4.1 Training the classifier

To train the ECLAIR ensemble classifier (Supplementary Fig. 4a), we first selected a training and testing set of interactions with known interface residues. We selected 400 interactions for each set, each composed evenly of homodimers and heterodimers (Supplementary Table 3). Importantly, we allowed no repeated proteins and no homologous interactions between the training and testing sets. We disallow repeated proteins to ensure that we do not report inflated performance metrics based upon learning shared interfaces of proteins with multiple partners. Also, since we use homology-based interaction models in lieu of our classifier whenever available, our exclusion of homologous interactions ensures that our classifier performance is measured assuming no homology templates are available, thereby accurately representing the type of information that will be available for the set of interactions for which we will predict.

In the first step of training we selected a set of information-rich features from each of the five feature categories that will be used to build the final classifiers. We first applied the feature engineering techniques previously described, aggregation and scaling, on each of the raw features. We then tested combinations of engineered features in order to select a high-performing set of engineered features to represent each category. Due to their established track record of success for predicting interface residues and in solving other biological problems, we measured performance of random forest classifiers trained on each set of features and optimized over residue window sizes and algorithm hyperparameters.

As with any machine learning classifier, hyperparameters dictate the behavior of the random forest classifier and can ultimately determine its success or failure for a given prediction task. For instance, using the default parameters encoded in popular implementations of random forest vastly undercuts performance for some classification problems, and selecting hyperparameters by hand is tedious and may introduce biases such as overfitting and cross-contamination of training and testing sets that invalidate performance metrics.

One popular automated approach for choosing hyperparameters is grid search, which, when employed properly during cross-validation, can largely avoid these pitfalls. However, grid search can be prohibitively expensive to perform at appropriate levels of granularity required to search a full hyperparameter space. Randomly ordered grid search or random continuous search may slightly improve the time needed to reach a near optimal set of hyperparameters. However for hyperparameter spaces consisting of more than 2 hyperparameters being tuned concurrently, a more directed approach is needed (Supplementary Fig. 4b).

Recently there have been advances in Bayesian methods for selecting optimal hyperparameters at lower computational cost. During cross-validation, these methods begin by sampling a hyperparameter space according to a pre-defined random distribution (i.e. normal or uniform) and then
selectively sample areas of the hyperparameter space that minimize a cost function. We used a recent method, the tree-structured Parzen estimator approach (TPE)\textsuperscript{84}, which allowed us to simultaneously tune up to 8 hyperparameters for each random forest, including the size of residue windows over which features are included (Supplementary Fig. 4c).

We selected combinations of engineered features to include in final classifiers by training a set of preliminary classifiers on candidate combinations of features. We used TPE to optimize each of the preliminary classifiers, and selected the features from the highest performing classifier to represent each of the 5 feature categories (Supplementary Fig. 4d, Supplementary Table 4). In this procedure (cross validation I), three-fold cross validation on the training set of interactions was used to evaluate each candidate set of hyperparameters, with a loss function calculated based on the area under the receiver operating characteristic (ROC) curve for the left-out folds.

We then assembled the final ensemble of classifiers, which is designed to cover common feature availability scenarios and therefore allow prediction on any interaction using all available features without introducing bias (as previously described). In total, the ensemble contains 8 random forest classifiers built from the top-scoring features from cross validation I. During training of each of these 8 classifiers, cross-validation was again performed with TPE, allowing hyperparameters and window sizes to be set to their optimal values, separate of those found in the previous round of cross-validation (Supplementary Fig. 4a).

For a single classifier of the ensemble (Classifier 2, trained on biophysical and conservation-based features), we also performed the three other aforementioned strategies for hyperparameter tuning (Supplementary Fig. 4e). Importantly, for sampling a minimal set of 3 potential values for each of the random forest hyperparameters, this required testing 648 combinations of hyperparameters for grid search-based strategies. To optimize Classifier 8 (trained on all 5 feature categories) with a more reasonable (though still restrictive) 10 potential values for each random forest hyperparameter would require >1.5 million rounds of cross-validation. Not only does TPE find a better combination of hyperparameters in fewer trials, it is able to sample the full continuous range of each hyperparameter, making it the only feasible solution for searching higher-dimensional search spaces across many classifiers.

4.2 Evaluation of the ensemble
The performance of each of the individual 8 sub-classifiers was evaluated on the previously untouched 400 interactions in the testing set. As expected, the performances of the sub-classifiers increase as more informative features are added, with the area under the ROC curve ranging from ~0.6 for the classifier trained on the fewest features to ~0.8 for the classifier trained on all features (Supplementary Fig. 6). We
also note that the distribution of raw probabilities returned by each sub-classifier is expected based on the type of features that were used. For instance, sub-classifiers 1-3, which were trained using only sequence-based biophysical features, produce roughly normal distributions peaking around a probability of ~10% (the true incidence of interface residues in our testing set). Classifiers 4-8, which incorporate structural features, produce distributions that are bimodal, showing the distinguishing power of using known solvent-accessible surface area as a feature input (residues that are definitely buried will be predicted to have a roughly 0% chance of being at the interface). However, for the interpretation of results from different classifiers, we note that each classifier produces probabilities that are approximately true (Supplementary Fig. 6d). This means that no matter which classifier produces a 30% interface probability prediction, it is expected that ~30% of such residues will actually be at the interface. This enables us to use different classifiers to predict for different residues with the expectation that the raw returned probabilities are comparable, and forms the basis for our use of 8 independent classifiers as an ensemble. It is worth noting that we could only use predictions from different classifiers interchangeably because we used a single testing and training set that produced classifiers with measurable and comparable output probabilities. This would not be the case for combining predictions from separate, pre-existing methods that have not been trained and benchmarked in a unified manner.

For reporting performance metrics at various discrete and comparable levels of confidence, we consider evenly separable fifths of our raw prediction scores, constituting Very Low, Low, Medium, High, and Very High prediction confidence. We find that each higher tier of predictions outperforms the previous tier using a community benchmark in measures of true positive rate, false positive rate, specificity, and precision (positive predictive value) (Supplementary Table 7).

4.3 Classification of unknown interfaces with systematic computational and experimental evaluation

After performance evaluation, the 8 sub-classifiers were finally retrained on the entire set of known interface residues, including both the training and testing sets, a common practice in machine learning that makes full use of labeled data to build a final model. Typically, this leaves no remaining data on which to test the final model, however, by evaluating on a set of 190 interactions whose crystal structures were deposited in the PDB after the initial submission of this study, but before publication, we were able to compare the performance of the training-set only model to the fully-trained model. We find that our finally trained model performs slightly better than the initial model (Supplementary Table 8), justifying the use of the fully-trained model in our final prediction of unknown interfaces.

The fully trained ensemble classifier was used to predict interface residues in each of 185,957 interactions without known experimental structures or homology models. As expected, the majority of
residues are predicted to be away from the interface (only ~12% of residues in our training set are at the interface), and appropriately few residues are predicted with “High” and “Very High” confidence (Supplementary Fig. 5a). Importantly, the distribution of predicted interface residues ensures good coverage of the interactomes; >50% of interactions have at least 10 residues predicted with “High” or “Very High” confidence to be at the interface, and >99% interactions have at least one Medium confidence prediction, making the dataset widely applicable to all interactions (Supplementary Fig. 5b).
Supplementary Note 5: Training classifiers with incomplete data

The problem of building machine learning classifiers based on feature sets with missing values is widespread in biology, where datasets are often incomplete. For predicting interface residues, it has proven especially problematic as many of the most informative features, such as structurally-derived features, are available for only a fraction of interactions.

A popular approach to address the missing feature problem is imputation. Imputation replaces missing values in a feature array with substitutes intended not to influence the result of the classifier, such as the mean or mode of the feature across the dataset so that machine learning algorithms can operate as designed. Most commonly used imputation techniques (including mean, mode, and regression imputation) operate under the assumption that features are missing at random with respect to data labels in a classification problem. However, in many cases, including predicting interface residues, this assumption is violated. For instance, residues missing structural features (such as solvent-accessible surface area and docking results) are significantly more likely also to be missing in co-crystal structures from which we derive our interface residue truth set for training, due to bias in PDB structures. Therefore, residues missing structural features will be classified almost exclusively as non-interface as there will be very few examples of interface residues outside of crystalized protein regions during training. By employing an ensemble of fully trained classifiers covering different feature-availability scenarios, we ensure that our classifiers are neither explicitly nor implicitly trained to predict based on the missingness of features, which, in the case of predicting for residues lacking structural evidence, would introduce a severe bias.

Ensemble classifiers are methods in which the outputs of many classifiers are combined to produce a single prediction. They have most commonly been used to improve the result of a single predictor by combining its results with other pre-existing predictors. They can also be used ab initio to train a set of classifiers on subsets of a complete feature set to improve classification performance and reduce noise, such as in boosting and bagging. More recently, ensemble classifiers have also been used to address the missing feature problem, for instance by building classifiers trained on random selections of features and employing a voting system of suitable classifiers during prediction. However, such approaches can be computationally expensive to perform, especially for high-dimensional data with a large percentage of missing features as this requires a large number of classifiers to cover all potential patterns in which features can be missing.

For the prediction of interface residues, features are absent or present in co-dependent, all-or-none patterns (i.e. a residue is likely to have all or no structure-based features available), which allows us to train a simplified ensemble of classifiers that cover common feature spaces. This model alleviates any need for imputation as all residues are predicted using the complete set of features on which a single sub-
classifier of the ensemble was trained. Furthermore, training this smaller set of classifiers vastly improves
the computational performance of the algorithm and makes the results more interpretable as we know
which classes of features were used in each prediction and the performance of our testing set on each
classifier.
Supplementary Note 6: Utility of Interactome INSIDER for genomic studies

We have demonstrated the utility of interface predictions in Interactome INSIDER for genomic studies by assessing well-known genomic properties of these predictions compared to known interfaces. For instance, our predicted interfaces are similarly enriched for human disease mutations as known interfaces (Fig. 4a), and like known interfaces, our predicted interfaces are enriched for recurrent cancer mutations (Fig. 4b) and harbor more rare and deleterious population variants compared to outside of interfaces (Fig. 4c-e). Furthermore, we performed two pilot studies to show how our predicted interfaces could be used alongside disease and pathway information to implicate new proteins in disease and hypothesize likely mechanisms of their action (Fig. 5a and Fig. 6).

Interactome INSIDER contains predicted interfaces for many more interactions than other benchmarked methods (in Supplementary Table 1) can be applied to, making it especially useful for genomic studies. Due to availability of rich structural features, predicted interfaces in interactions that cannot be predicted using other methods will be largely based on sequence-derived features in Interactome INSIDER. While predictions based on structural features are undoubtedly of high quality, sequence-based predictions can still be applied to genomic studies. For instance, we show that ECLAIR-predicted interface residues in human interactions lacking structural features are still enriched for disease mutations (Supplementary Fig. 8a), recurrent cancer mutations (Supplementary Fig. 8b), and rare and deleterious population variants (Supplementary Fig. 8c-e) at similar rates to known interface residues derived from crystalized co-bound proteins. This suggests that use of any of the interfaces in Interactome INSIDER is likely to yield biologically meaningful trends.

Studying interaction interfaces at the domain level is also an accepted strategy, since it is well known that proteins largely function through evolutionarily conserved domains\textsuperscript{52,90,91}. When high confidence predictions are sparse, often interacting domains can still be identified by the presence of many lower confidence predictions in close proximity and localized to domain regions. We find that our predicted interface domains also display the same genomic properties we had calculated for individual residues. For instance, we show that ECLAIR-predicted interface domains in human interactions are enriched for disease mutations (Supplementary Fig. 9a), recurrent cancer mutations (Supplementary Fig. 9b), and rare and deleterious population variants (Supplementary Fig. 9c-e) at similar rates to known interface domains derived from crystalized co-bound proteins. This suggests the validity of using predicted interface domains in genomic studies.

While the precision of ECLAIR to predict interface residues may be slightly lower for predictions made without structural features, interface domain prediction precision is quite high even for predictions made with only sequence-based features (Supplementary Fig 10a-c). ECLAIR predicts interface residues to be within domains, thereby identifying interface domains, at a similar rate regardless of
whether structure- or sequence-based features are used for prediction (Supplementary Fig 10d). Importantly, interface domains derived from co-crystal structures, predicted using structural features, and predicted with sequence-based features all display marked enrichment for disease mutations (Supplementary Fig 10e), suggesting their utility in genomic studies.
Supplementary Note 7: Mutagenesis experiment methods

7.1 Choosing mutations for mutagenesis

From the Exome Sequencing Project, we randomly selected 2,164 population variants. These population variants were distributed among known interfaces and known non-interfaces (for proteins with co-crystal structures) and in each of our top four predicted interface confidence categories: Low, Medium, High, and Very High. We then introduced each of these mutations via our mutagenesis pipeline described below and observed the fraction of mutations that disrupted interactions in each predicted category and in known and known-non interfaces.

Different numbers of mutations were tested in each category, varying based on number of actual predictions made in each category, and this is taken into account intrinsically via error bars and in statistical significance calculations (Z-test) in Figure 2c. The breakdown of mutations tested in each category is: Low (1,270), Medium (267), High (79), and Very High (48). Additionally, 360 variants in known interfaces and 140 variants in known non-interfaces were also tested.

7.2 Construction of mutant alleles using high-throughput site-directed mutagenesis PCR

Mutagenesis experiments were carried out based on the protocol developed before. Briefly, primers for site-directed mutagenesis were selected based on a customized version of the protocol accompanying the Stratagene QuikChange Site-Directed Mutagenesis Kit (200518). All wild-type clones were obtained from the human ORFeome v8.1 collection. To generate mutant alleles, sequence-verified single-colony wild-type clones and their corresponding mutagenic primers were aliquoted into individual wells of 96-well PCR plates. Mutagenesis PCR was then performed as specified by the New England Biolabs (NEB) PCR protocol for Phusion polymerase (M0530L), noting that PCR was limited to 18 cycles. The samples were then digested by DpnI (NEB R0176L) according to the manufacturer’s manual. After digestion, 10 µL of mutagenesis PCR products were transformed into 100 µL of competent E. coli. Finally, colony picking was done using four-sector agar plates (VWR 25384-308) that were partitioned into four non-contacting quadrants with glass beads poured onto each plate quadrant. Each bead-filled quadrant was inoculated with ~50 µL of transformed bacteria. This was then spread by lightly shaking the four-sector agar plate. Our optimized transformation protocol results in a large number of well-separated single colonies that can be easily picked the next day. Upon recovery, single colonies from each quadrant were then picked and arrayed into 96-deep well plates filled with 300 µL of antibiotic media. Four colonies per allele were picked for next-generation sequencing.
7.3 DNA library preparation for Illumina sequencing

DNA library preparation was performed using NEBNext DNA Library Prep Master Mix Set for Illumina (NEB E6040S) according to the manufacturer’s manual. Briefly, 5 µg of pooled plasmid DNA (~100 µL, all samples were normalized to the same concentration) was sonicated to ~200 bp fragments. The fragmented DNA was first mixed with NEBNext End Repair Enzyme for 30 mins at 20 °C. Blunt-ended DNA was then incubated with Klenow Fragment for 30 mins at 37 °C for dA-Tailing. Subsequently, NEBNext Adaptor was added to dA-Tailed DNA. Adaptor-ligated DNA (~300 bp) was size-selected on a 2% agarose gel. Size-selected DNA was then mixed with one of the NEBNext Multiplex Oligos (NEB E7335S) and Universal PCR primers for PCR enrichment. At each step, DNA was purified using a QIAquick PCR purification kit (Qiagen 28104). Multiplexed DNA samples were combined and analyzed in one lane of a 1×100 bp run by Illumina HiSeq 2500.

7.4 Identifying successful instances of site-directed mutagenesis based on next-generation sequencing

The mutant colonies were barcoded and pooled. The multiplexed colonies were then run on an Illumina sequencer to give 1×100 bp reads. These reads were then de-multiplexed and mapped to the genes of interest using the BWA “aln” algorithm. For each allele, we identified all reads that mapped to the position of the mutation of interest (R_all) and those that actually contained the desired mutation (R_mut). We then calculated a normalized score (S) that quantifies the fraction of reads containing the desired mutation:

\[ S = \frac{\frac{1}{k} R_{mut}}{R_{all}} = \frac{k \times R_{mut}}{R_{all}} \]

where k is the number of different mutations in the same gene. We require S > 0.8 for a colony to be scored as containing the desired mutation.

7.5 Identifying unwanted mutations

One major advantage of our Clone-seq pipeline over traditional site-directed mutagenesis protocols using Sanger sequencing is that we can now carefully examine whether there are other unwanted mutations inadvertently introduced during the PCR process, in comparison with the corresponding wild-type alleles. It is essential to use clones with no unwanted mutations for downstream experiments, as the presence of
these will make it impossible to determine whether the observed disruption is due to the desired or other undesirable mutation(s).

We use samtools “mpileup”\(^7\) to obtain read counts for different alleles at each nucleotide for all the clones. We calculate the background sequencing error rate by calculating the average fraction of non-reference alleles across all nucleotides where we did not attempt to introduce a mutation. Any site that has a significantly higher fraction of non-reference alleles (using a \(P\) value cutoff of 0.2 from a cumulative binomial test) is considered to have an unwanted mutation. A lenient \(P\) value cutoff (0.2 as opposed to the more traditionally used 0.05 or 0.01) implies more stringent filtering in this case because we want to eliminate type II errors i.e., we want to identify all unwanted mutations at the cost of discarding a few clones that actually do not have any unwanted mutations.

7.6 Y2H assay
Y2H was performed as previously described\(^5\). All wild-type/mutant clones were transferred by Gateway LR reactions into our Y2H pDEST-AD and pDEST-DB vectors. All DB-X and AD-Y plasmids were transformed individually into the Y2H strains \(MAT\alpha\) Y8930 and \(MAT\alpha\) Y8800, respectively. Each of the DB-X \(MAT\alpha\) transformants (wild-type and mutants) were then mated against corresponding AD-Y \(MAT\alpha\) transformants (wild-type and mutants) individually using automated 96-well procedures, including inoculation of AD-Y and DB-X yeast cultures, mating on YEPD media (incubated overnight at 30 °C), and replica-plating onto selective Synthetic Complete media lacking leucine, tryptophan, and histidine, and supplemented with 1 mM of 3-amino-1,2,4-triazole (SC-Leu-Trp-His+3AT), SC-Leu-His+3AT plates containing 1 mg/l cycloheximide (SC-Leu-His+3AT+CHX), SC-Leu-Trp-Adenine (Ade) plates, and SC-Leu-Ade+CHX plates to test for CHX-sensitive expression of the \(LYS2::GAL1\)-\(HIS3\) and \(GAL2\)-\(ADE2\) reporter genes. The plates containing cycloheximide select for cells that do not have the AD plasmid due to plasmid shuffling. Growth on these control plates thus identifies spontaneous auto-activators\(^9\). The plates were incubated overnight at 30 °C and “replica-cleaned” the following day. Plates were then incubated for another three days, after which positive colonies were scored as those that grow on SC-Leu-Trp-His+3AT and/or on SC-Leu-Trp-Ade, but not on SC-Leu-His+3AT+CHX or on SC-Leu-Ade+CHX.

Disruption of an interaction by a mutation was defined as at least 50% reduction of growth consistently across both reporter genes, when compared to Y2H phenotypes of the corresponding wild-type allele as benchmarked by 2-fold serial dilution experiments. All Y2H experiments were repeated 3 times.

7.7 Immunoblotting of WT and mutant (Y353S) SMAD4
WT and mutant (Y353S) SMAD4 were PCR cloned into the mammalian expression vector pcDNA3.1 (Invitrogen Life Technologies) using XbaI and NotI restriction sites. Flag-tag was introduced into the C-
terminal end. HEK293T cells, maintained in complete DMEM supplemented with 10% FBS, were transfected with these constructs and the empty vector using Lipofectamine 2000 reagent (Invitrogen) at a 6:1 (liter/gram) ratio with DNA. Cells were harvested 24 h after transfection. Transfected cells were gently washed three times in PBS and then resuspended using 200 μl 1% NP-40 lysis buffer (1% Nonidet P-40, 50 mM Tris-HCl pH 7.5, 150 mM NaCl, 1 EDTA-free Complete Protease Inhibitor tablet (Roche), 1 M sodium orthovanadate, 1 mM sodium fluoride) per well of 6-well plate for 20 min on ice in Eppendorf tubes. Extracts were cleared by centrifugation for 10 min at 13,000 r.p.m. (>16,000g) at 4 °C. Protein lysate (20 μl) were subjected to SDS-PAGE and protein blotting. Anti-Flag (Sigma-Aldrich), anti-γ-tubulin (Sigma-Aldrich T5192) antibodies were used for immunoblotting analyses.
References


