Electronic Companion to "Harmonizing Safety and Speed: A Human-Algorithm Approach to Enhance the FDA's Medical Device Clearance Policy"

by Mohammad Zhalechian, Soroush Saghafian, and Omar Robles

This electronic companion comprises all the proofs of our theoretical findings, along with supplementary details and additional results from computational experiments.

Appendix A: Text Mining Algorithm

In this section, we describe the process of mining the predicate data for devices cleared through the 510(k) pathway. It consists of five main steps, each of which is described in detail below.

Step 1: Downloading the 510(k) files of cleared devices. The FDA maintains downloadable files containing information about devices cleared under the 510(k) process from 1976 to the present. These files are updated monthly, typically on or around the 5th of each month. Each record includes the submission identifier (the 510(k) identifier), along with other applicant and device information such as the applicant's name, FDA decision date, and product code. However, predicate information is not included. These files are publicly available without restriction on the FDA website. We merged all available files to create a comprehensive list of 510(k) devices cleared between 1976 and 2020, storing the information in a CSV file.

Step 2: Downloading the summary document for each cleared device. Predicate information for each cleared device is typically available in a summary document housed within the FDA data warehouse. Unlike the 510(k) files of cleared devices, these summary documents are not available for bulk download. Instead, each device has a webpage that contains a link to the summary document for that device.

The device webpage is accessible through an FDA searchable database. Figure EC.1 shows an example search for device K192656 using the user interface.

The search results, as shown in Figure EC.2, contain much of the same information available in the downloadable files from Step 1, with one notable exception: the webpage includes a link labeled "Summary," which typically contains predicate information for that device. Each device's webpage follows a similar URL structure, differing only in the unique 510(k) identifier assigned to each submission.

To automate the retrieval process and eliminate the need for labor-intensive manual searches, we developed a Python algorithm that uses the CSV file generated in Step 1 to download each device's respective summary document. The algorithm uses the "requests", "BeautifulSoup", and "urllib.request" libraries to iteratively access the URL, search for a link containing the term "Summary," and download the embedded document. The document is then saved as a PDF, and the CSV file is updated to indicate whether a document was successfully downloaded using the 510(k) identifier as a reference.

Step 3: Scraping the predicate identifier from each summary document. The summary documents vary in format and quality. Some of them are not readily searchable using optical character recognition

(OCR), some lack predicate 510(k) identifiers but provide predicate names, and some contain no predicate information. To address these challenges, we developed a mechanism within our algorithm to handle certain issues automatically, while others are resolved manually.

In general, the algorithm iteratively searches for predicate 510(k) identifiers within a summary document and records them in a CSV file. It utilizes the following main libraries: "pdfplumber", "Image", "pyocr". The algorithm processes the PDF documents downloaded in Step 2. For each PDF document, it searches for a predicate identifier, which follows a standard format consisting of the letter "K" or "P" followed by six digits (e.g., K032245, P230042). Devices cleared through the 510(k) pathway have identifiers that start

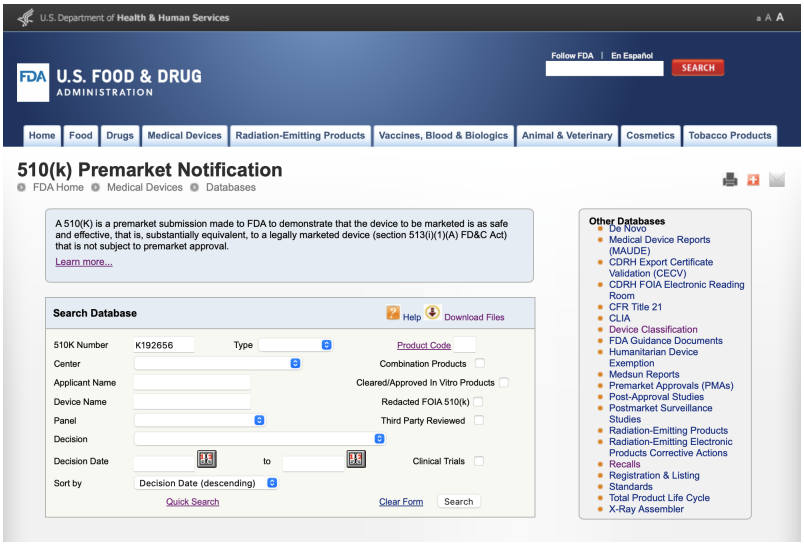

Figure EC.1 FDA 510(k) Device Search Interface

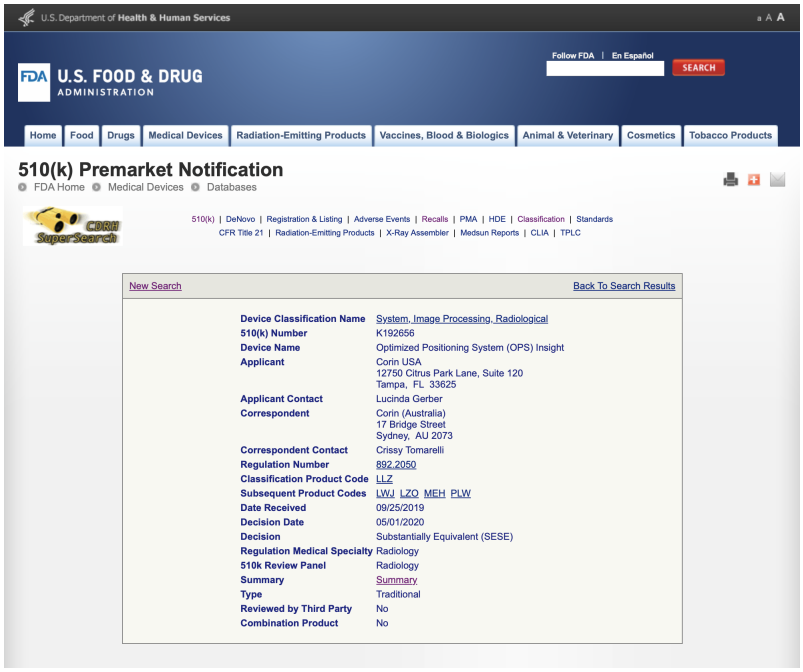

Figure EC.2 Search results for device K192656

with the letter "K", while those cleared through the premarket approval (PMA) pathway have identifiers that start with the letter "P".

Once an identifier is found, the algorithm records it in the CSV file associated with the device's summary document. If the algorithm fails to extract at least one predicate identifier from a summary document, it converts the document into an image for OCR processing before making a second attempt to locate and record the predicate identifier(s). If the algorithm is unable to retrieve a predicate identifier from the summary document, it records a blank space in the CSV file.

Step 4: Manually updating missing predicate identifiers. In cases where the algorithm fails to record at least one predicate identifier, we manually review the summary document and record the predicate identifier(s) if available. Occasionally, the summary document may list the descriptive name(s) of the predicate device(s) but not the predicate identifier(s). In a smaller fraction of instances, the predicate identifier(s) may contain typographical errors, such as a missing digit or an extra space (e.g., K12345 instead of K12 3456). In either case, we search for predicate device information manually.

To ensure confidence in the identification of predicate devices, we apply the following criteria: (1) the name (or identifier) of the predicate device must match the name (or identifier, excluding any typographical errors) in the summary document, and (2) the predicate device must have been cleared prior to the device for which it serves as a predicate. This approach prevents the misidentification of predicate devices that are different (e.g., newer models within a product line).

Although predicate information is a requirement in the FDA 510(k) application, it is not always included in the summary document. In these cases, the summary document only includes a general statement that the device is substantially equivalent to a previously cleared or approved device without specifying the exact predicate device. Fortunately, in the vast majority of cases, the summary document contains predicate information, which allowed us to identify the relevant device.

Step 5: Manually updating predicate identifiers when more than 10 predicates were recorded. In cases where the algorithm recorded more than 10 predicates, we manually review the summary document. Although no errors were found in this process, we would update the record of predicate identifier(s) had an error occurred.

While we conducted spot checks to ensure the correctness of identified predicates, we acknowledge that there is a chance that some additional predicates may have gone undetected in a small number of cases.

Appendix B: Variable Definitions

The FDA classifies each device into several medical specialties, such as Orthopedic (OR) or Cardiovascular (CV). *Medical Specialty* refers to the medical specialty of the applicant device identified by the FDA. The FDA classifies medical devices into three classes based on their risks and regulatory controls. Class I devices pose the lowest risk to patients, while Class III devices pose the highest risk. *Device Class* indicates the FDA device class. Most devices in the 510(k) program fall into Class I or Class II. A limited number of applicant devices (preamendments devices) were initially regulated as Class III with the intent that either the FDA would reclassify the device into a lower class or call for the premarket approval application. Due to a limited number of such applicant devices and the lack of a standard protocol, we only included applicant devices of

Class I or II in our analyses. *Country Code* indicates the country of origin for a device manufacturer. We reduced the number of country code levels by preserving the most common ones and re-coding the others with a frequency of less than 1% as "Other." The Center for Devices and Radiological Health (CDRH) associates each medical device with a *Product Code* based upon the medical device function. For example, ultrasonic pulsed doppler devices are assigned to "IYN," while Ultrasonic Pulsed Echo imaging devices are assigned to "IYO." We reduced the number of product code levels for each medical specialty by preserving the most common ones and re-coding the others with a frequency of less than 5% as "Other Medical Specialty." *Implantable* is a flag indicating whether a device is designed to be placed into a surgically or naturally formed cavity of the human body. *Life Sustaining/Supporting* is a flag indicating whether a device is essential to restoring or continuing a bodily function.

In the FDA 510(k) program, the *substantial equivalence* is often evaluated based on the similarities between the predicate devices and the applicant device in terms of the composition and design (Zuckerman et al. 2014). In the publicly available datasets, the information on predicate devices is not directly linked to the corresponding applicant devices. Thus, we created several variables corresponding to predicate devices, which can be classified into the following three categories.

The first category accounts for the similarity between an applicant device and predicate devices. *Number of Predicates* in this category is a count of the number of predicate devices identified for an applicant device. *Prop. of Unmatched Medical Specialties* measures the ratio of the number of distinct medical specialties among an applicant device's predicates that differ from the applicant device's own specialty, divided by the total number of predicate devices. Similarly, *Prop. of Unmatched Product Codes* measures the ratio of the number of distinct product codes among an applicant device's predicates that differ from the applicant device's own product code, divided by the total number of predicate devices.

The second category accounts for the age of predicate devices. *Predicate Average Age* is the difference between the average year of approval of predicate devices and the application submission date of the applicant device. *Predicate Median Age* is the difference between the median year of approval of predicate devices and the application submission date of the applicant device. *Predicate Newest Age* indicates the difference between the year of approval of the newest predicate device and the application submission date of the applicant device. Similarly, *Predicate Oldest Age* indicates the difference between the year of approval of the oldest predicate device and the application submission date of the applicant device.

The third category accounts for the recall status of predicates. *Number of Class 1 Recalls* is a count of the total number of Class 1 recalls among the predicate devices identified for an applicant device. *Number of Class 2 Recalls* and *Number of Class 3 Recalls* can be defined similarly. *Variance of Recalls* is the sample variance of the number of recalls of predicate devices. For example, consider an applicant device with three predicate devices. In the first scenario, the first predicate has six recalls, while the second and third predicates have zero recalls. In the second scenario, each predicate device has two recalls. The sample variance of recalls for the first scenario is 12, while it is zero for the second scenario. Another interesting piece of information to consider is the timing of the recall events. For example, the importance of a recall event of a predicate device that has occurred many years prior to an applicant device's submission date may differ from a recent

recall event. *Weighted Recall Score* is calculated as the weighted number of recalls across all predicate devices associated with a given submission, divided by the total number of recalls for the corresponding predicate devices. We define a time window of ten years such that a recall event is negligible if it occurred ten or more years before the applicant device's submission date. For the other recall events, we assign weights based on the time difference. Thus, a recall event that has occurred at the submission date receives a weight of one, and a recall event that has occurred ten years before the submission date is assigned a weight of zero. For recall events that fall within this ten-year window, weights are calculated proportionally based on their distance from the submission date (e.g., a recall event that happened 4 years ago would be assigned a weight of 0.6).

Overall, we constructed 17 predictors for each applicant device, including 11 continuous measures, 3 binary indicator flags (Device Class, Implantable, and Life-Sustaining/Supporting), and 3 multi-level categorical variables (Medical Specialty, Product Code, and Country Code). During data preprocessing, we applied one-hot encoding to the multi-level categorical variables, resulting in 20 unique categories for Medical Specialty, 111 for Product Code, and 12 for Country Code.

Appendix C: ML Models Implementation and Cross-Validation Procedures

In this section, we outline the key implementation details of our ML models and cross-validation (CV) strategy. All models were implemented in Python using open-source packages.

Data Splitting and Preprocessing

We used a stratified random split to divide the full dataset into 70% training and 30% test data, ensuring the recall rate is preserved across splits. This was implemented using the `train_test_split()` function from the scikit-learn library.

On the training set, stratified 10-fold cross-validation was used to tune hyperparameters, implemented via the `StratifiedKFold` function also from the scikit-learn library. Continuous variables were standardized based on training data statistics.

ML Models

All models were evaluated using mean AUC over 10-fold CV, repeated 10 times with shuffled folds.

- **Logistic Regression (Lasso and Ridge):** Implemented using `sklearn.linear_model.LogisticRegression` with `liblinear` solver. Penalty type (`l1` for Lasso, `l2` for Ridge) and regularization parameter `C` were tuned.
- **Decision Tree Classifier:** Implemented using `sklearn.tree.DecisionTreeClassifier`. Hyperparameters `min_samples_split` and `max_depth` were tuned.
- **Random Forest Classifier:** Implemented using `sklearn.ensemble.RandomForestClassifier`. Hyperparameters `n_estimators` and `max_depth` were tuned. For improved performance and to enable parallelism, we set `max_features = sqrt` and `n_jobs = -1`.
- **Gradient Boosting Classifier:** Implemented using `sklearn.ensemble.GradientBoosting Classifier`. Hyperparameters `n_estimators` and `learning_rate` were tuned. We fixed `max_depth = 4` and `subsample = 0.8`.

- **Cox Proportional Hazards Model (Lasso and Ridge):** Implemented using `sklearn_survival.CoxnetSurvivalAnalysis`. Regularization parameter `alpha` was tuned. For Lasso, `l1_ratio = 1.0` and for Ridge, `l1_ratio = 1e-5`.

Appendix D: Robustness Checks for Censoring Effect

Recall that in our ML models, the primary outcome is a binary recall event indicating whether an applicant device had at least one recall between its clearance date and the end of our observation window. We include all devices cleared from 2008 to 2020 and we track their recall events until the end of 2021. This could create a censoring effect for the most recent applicant devices. That is, the devices approved at dates closer to the end of the observation period may have a lower chance of experiencing a recall than those approved at the beginning of the observation period. In this section, we employ two robustness checks to assess the impact of censoring.

In our first method, we apply survival analysis techniques that are commonly used in the literature in the presence of censored outcomes. The Cox proportional hazards model is a widely used survival analysis technique known for its semi-parametric nature and interpretability. It models the hazard rate for an event as a linear combination of covariates, allowing the coefficients to be interpreted in terms of hazard ratios. However, standard Cox models can struggle with correlated feature spaces, which leads to numerical instability when inverting covariance matrices. To address this and avoid overfitting, we employ regularization techniques by incorporating Lasso and Ridge penalties into the Cox model.

We train two regularized Cox models with Ridge and Lasso penalties using scikit-survival, which is a Python module for survival analysis built on top of the scikit-learn library. Penalty parameters are optimized through a 10-fold cross-validation procedure based on the area under the curve (AUC) metric. Table EC.1 reports the predictive performance of the regularized Cox models on unseen 510(k) applicant devices. The cross-validated AUC values for the two Cox models range from 0.74 to 0.77. Among them, the Cox model with Ridge penalty demonstrates a higher AUC with lower variance. However, it still underperforms compared to gradient boosting, which remains our selected model for recall risk prediction.

Our second method involves an incremental analysis to systematically assess the impact of censoring. In this approach, we construct train and test datasets where devices have sufficient follow-up time to capture potential recall events. This setup enables us to analyze how varying levels of censoring in data affect model performance. Figure EC.3 shows the distribution of recall time for all devices in our dataset that have been recalled. The bars show the absolute percentage of recalls after years from submission, and the red curve shows the cumulative recall percentage up to different years after submission. The figure shows that a substantial portion of recalls occurs relatively early, where 71.9% of recalls happen up to 4 years after submissions.

Table EC.1 Performance comparison of survival models

Model	CV-AUC (95% CI)	OOS-AUC (95% CI)
Cox with Ridge penalty	0.77 (0.73, 0.81)	0.76 (0.75, 0.77)
Cox with Lasso penalty	0.74 (0.70, 0.78)	0.74 (0.72, 0.75)

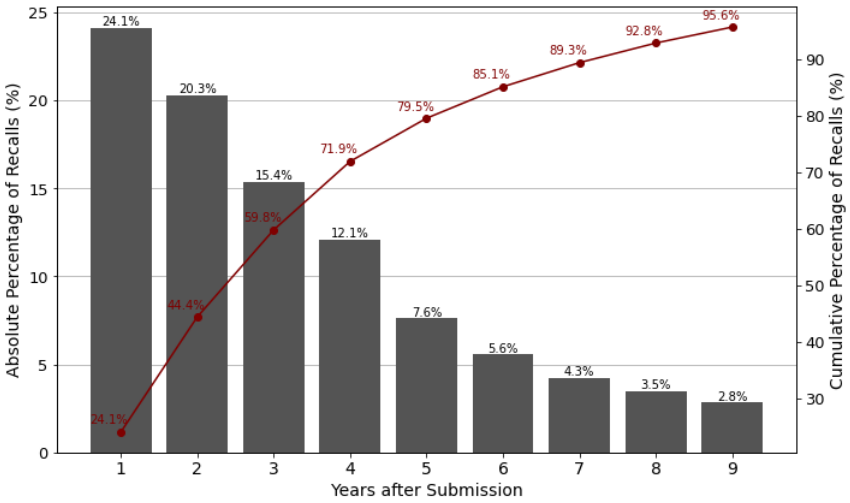

Figure EC.3 Distribution of recall time

Table EC.2 Area under the curve across follow-up times

Years After Submission	CV-AUC (95% CI)	OOS-AUC (95% CI)
≥ 9	0.75 (0.70, 0.80)	0.74 (0.72, 0.76)
≥ 8	0.75 (0.71, 0.80)	0.74 (0.73, 0.76)
≥ 7	0.75 (0.71, 0.80)	0.76 (0.75, 0.78)
≥ 6	0.75 (0.71, 0.80)	0.76 (0.75, 0.78)
≥ 5	0.75 (0.71, 0.79)	0.76 (0.75, 0.78)
≥ 4	0.76 (0.73, 0.80)	0.75 (0.74, 0.77)

In our incremental analysis, we consider follow-up thresholds of at least 9, 8, 7, 6, 5, and 4 years, allowing us to systematically assess how varying levels of censoring in data impact model performance. For example, a threshold of 9 years in this analysis means that we only include devices that have had at least 9 years of follow-up in our data. According to Figure EC.3, 95.6% of these devices should have already experienced a recall event (if applicable), which ensures minimal censoring in the dataset. By gradually reducing the follow-up threshold, we assess whether training with progressively censored data impacts model predictions. Table EC.2 reports the CV-AUC of our selected gradient boosting model across different follow-up thresholds. The CV-AUC values range from 0.75 to 0.76 and the OOS-AUC values range from 0.74 to 0.76, indicating that the model maintains stable performance across varying levels of data censoring. Comparing these results to the model's AUC using all data, we find that our ML model is robust to potential censoring effects introduced by more recent device approvals. Our additional analysis further confirms that the main findings regarding important variables are also robust to censoring (see Appendix E).

Based on our robustness checks, we find that the predictive performance of our selected model as well as the identification of key variables remain stable across different levels of data censoring. Thus, it provides a reliable approach for recall risk prediction.

Table EC.3 Output summary of the logistic regression model with selected variables by Log Reg with Lasso

Independent Variable	Coefficient	Standard Error	p-value
Weighted Recall Score	0.892	0.112	0.000*
Num. of Class 1 Recalls	-0.153	0.156	0.326
Num. of Class 2 Recalls	0.060	0.020	0.002*
Predicate Median Age	-0.022	0.012	0.066
Predicate Newest Age	-0.030	0.010	0.003*
Predicate Oldest Age	-0.001	0.007	0.910
Life Sustaining/Supporting	0.906	0.203	0.000*
Device Class	-0.880	0.217	0.000*
Implantable	-0.054	0.174	0.758
Prop. of Unmatched Specialties	-0.369	0.205	0.072
Prop. of Unmatched Product Codes	-0.056	0.124	0.653
Country Codes	Yes		
Product Codes	Yes		

Note: Significance code * indicates $p < 0.05$.

Appendix E: Robustness Check for Variable Significance

In this section, we aim to validate our insights on the important variables correlated with recall risk by examining the important variables in a logistic regression with Lasso regularization (Log Reg with Lasso). Our model comparison results (see Table 3) indicated that Log Reg with Lasso performs comparably to our primary model (gradient boosting), making it a useful alternative for validating our key findings. Additionally, we use Cox with Lasso (see Table EC.1) as another alternative to check whether our key findings are robust to potential censoring effects.

We directly assess the significance of the predictors retained in both models. The standard inference tools (e.g., p-values) are not directly applicable to regularized models. To address this, we adopt a post-Lasso (refit) approach with sample-splitting (Wasserman and Roeder 2009). Specifically, we fit each model on the training set to identify variables with non-zero coefficients, and then fit a logistic/Cox model with no regularization on the testing set using only these selected variables. Tables EC.3 and EC.4 report the coefficients, standard errors, and p-values from the refitted logistic regression and Cox proportional hazard models. According to Table EC.3, we find that all variables identified as important in the SHAP analysis of the selected model (gradient boosting) remain statistically significant, except for Predicate Median Age, Predicate Oldest Age, and Variance of Recalls. Although the p-value for Predicate Median Age is not statistically significant at the 0.05 level (p = 0.066), the variable still appears to be an important predictor and exhibits a negative association with the log-odds of recall. While Predicate Oldest Age is among the selected variables in Log Reg with Lasso, its p-value is not statistically significant. We also note that the Variance of Recalls was not retained by the Log Reg with Lasso model. Similar results are observed in Table EC.4 corresponding to the Cox with Lasso model.

Overall, the consistent direction and significance of major predictors (Weighted Recall Score, Number of Class 2 Recalls, Predicate Newest Age, Device Class, Life-Sustaining/Supporting) across both models

Table EC.4 Output summary of the Cox proportional hazard model with variables selected by Cox with Lasso

Independent Variable	Coefficient	Standard Error	p-value
Weighted Recall Score	0.893	0.069	0.000*
Num. of Class 1 Recalls	0.182	0.058	0.002*
Num. of Class 2 Recalls	0.067	0.009	0.000*
Predicate Median Age	−0.004	0.009	0.626
Predicate Newest Age	−0.033	0.008	0.000*
Predicate Oldest Age	0.008	0.005	0.109
Life-Sustaining/Supporting	0.902	0.098	0.000*
Device Class	−0.520	0.215	0.016*
Country Codes	Yes		
Product Codes	Yes		

Note: Significance code * indicates $p < 0.05$.

strengthen confidence in our main findings on the significant predictors of recall risk. Regarding the Predicate Median Age, Predicate Oldest Age, and Variance of Recalls, although they help the selected model to have a better prediction of recall risk, the results suggest that one needs to be careful about their association with recall risk in general.

Appendix F: Sensitivity Analysis of the Key Input Parameters of the Optimization Model

We investigate the impact of input parameters for our optimization model on various metrics, including the acceptance and rejection rates of both safe and unsafe devices, as well as the FDA's workload (currently stands at 100% as all devices are evaluated by the FDA's committees). We also measure the percentage point (pp) improvement over the FDA's current practice, which is defined as the difference between the FDA's current recall rate and the recall rate under our policy. This information can assist decision-makers in selecting the right input parameters that align with their criteria. The results for the training and test sets are summarized in Tables EC.5 and EC.6. In these tables, each input parameter is considered at three levels: low (L), medium (M), and high (H). For the parameters ξ^{ru} and ξ^{as}, we have $L = 0.3$, $M = 0.5$, and $H = 0.7$. For the parameter ρ, the values are $L = 0.4$, $M = 0.6$, and $H = 0.8$. We note that the "N/A" values in the last three rows indicate that the corresponding input parameters render the problem infeasible. The infeasibility of these cases stems from the fulfillment of condition (c) in Theorem 2.

As an example, consider the case where $\xi^{ru} = M$, $\xi^{as} = L$, and $\rho = M$ (i.e., M-L-M combination) in Table EC.5, which presents the metric values for the training set. We observe that using our policy results in rejecting 50.2% of unsafe devices and accepting 31.0% of safe devices. Among the accepted devices, 5.0% would experience a future recall, while 14.5% of the rejected devices would face no future recall. Accordingly, this policy leads to a 5.2 percentage point improvement (i.e., 50.2% improvement) in the recall rate compared to the FDA's current practice, which is 10.3%. Additionally, it results in a 46.6% reduction in the workload of the FDA's committees, as only 53.4% of the devices will be forwarded to the FDA's committees for in-depth evaluation, while the remaining 46.6% will be automatically accepted or rejected.

We observe substantial gains in certain scenarios, such as the L-L-L combination in Table EC.6, which results in a remarkable 60.1% workload reduction and a significant 6 percentage point improvement (i.e., 58.6% improvement). However, it is essential to note a vital caveat. Some combinations that offer significant improvements in workload reduction and recall rate also lead to an unreasonably high rate of rejection of safe devices. This phenomenon arises in the specific setting of low and high thresholds that, in certain cases, reject a substantial proportion of safe devices. Therefore, the selection of input parameters must strike a delicate balance between reducing workload and maintaining an acceptable rate of rejection for safe devices.

Table EC.5 Comparative analysis of key metrics (percentages) across combinations of input parameters in the training set

ξ^{ru}	ξ^{as}	ρ	Reject Unsafe	Accept Safe	Accept Unsafe	Reject Safe	Reject Rate	Workload	Absolute Imp. (pp)
L	L	L	66.2	31.0	5.0	28.1	32.1	39.6	6.8
L	L	M	41.3	31.0	5.0	8.3	11.7	60.0	4.2
L	L	H	30.1	31.0	5.0	3.5	6.2	65.4	3.1
L	M	L	42.2	51.2	15.6	8.9	12.3	40.1	4.4
L	M	M	30.1	51.2	15.6	3.5	6.2	46.2	3.1
L	M	H	30.1	51.2	15.6	3.5	6.2	46.2	3.1
L	H	L	30.1	70.7	32.2	3.5	6.2	27.1	3.1
L	H	M	30.1	70.7	32.2	3.5	6.2	27.1	3.1
L	H	H	30.1	70.7	32.2	3.5	6.2	27.1	3.1
M	L	L	66.2	31.0	5.0	28.1	32.1	39.6	6.8
M	L	M	50.2	31.0	5.0	14.5	18.2	53.4	5.2
M	L	H	50.2	31.0	5.0	14.5	18.2	53.4	5.2
M	M	L	50.2	51.2	15.6	14.5	18.2	34.2	5.2
M	M	M	50.2	51.2	15.6	14.5	18.2	34.2	5.2
M	M	H	50.2	51.2	15.6	14.5	18.2	34.2	5.2
M	H	L	50.2	70.7	32.2	14.5	18.2	15.1	5.2
M	H	M	50.2	70.7	32.2	14.5	18.2	15.1	5.2
M	H	H	50.2	70.7	32.2	14.5	18.2	15.1	5.2
H	L	L	71.1	31.0	5.0	32.8	36.8	34.9	7.3
H	L	M	71.1	31.0	5.0	32.8	36.8	34.9	7.3
H	L	H	71.1	31.0	5.0	32.8	36.8	34.9	7.3
H	M	L	71.1	51.2	15.6	32.8	36.8	15.7	7.3
H	M	M	71.1	51.2	15.6	32.8	36.8	15.7	7.3
H	M	H	71.1	51.2	15.6	32.8	36.8	15.7	7.3
H	H	L	"N/A"	"N/A"	"N/A"	"N/A"	"N/A"	"N/A"	"N/A"
H	H	M	"N/A"	"N/A"	"N/A"	"N/A"	"N/A"	"N/A"	"N/A"
H	H	H	"N/A"	"N/A"	"N/A"	"N/A"	"N/A"	"N/A"	"N/A"

Note: The recall rate of the FDA's current practice in the training set is 10.3%.

Table EC.6 Comparative analysis of key metrics (percentages) across combinations of input parameters in the testing set

ξ^{ru}	ξ^{as}	ρ	Reject Unsafe	Accept Safe	Accept Unsafe	Reject Safe	Reject Rate	Workload	Absolute Imp. (pp)
L	L	L	58.6	30.7	8.9	28.6	31.7	39.9	6.0
L	L	M	32.9	30.7	8.9	9.7	12.1	59.5	3.4
L	L	H	19.9	30.7	8.9	4.7	6.3	65.3	2.0
L	M	L	33.7	50.0	22.8	10.1	12.6	40.3	3.5
L	M	M	19.9	50.0	22.8	4.7	6.3	46.6	2.0
L	M	H	19.9	50.0	22.8	4.7	6.3	46.6	2.0
L	H	L	19.9	70.1	39.5	4.7	6.3	26.8	2.0
L	H	M	19.9	70.1	39.5	4.7	6.3	26.8	2.0
L	H	H	19.9	70.1	39.5	4.7	6.3	26.8	2.0
M	L	L	58.6	30.7	8.9	28.6	31.7	39.9	6.0
M	L	M	42.8	30.7	8.9	15.4	18.2	53.3	4.4
M	L	H	42.8	30.7	8.9	15.4	18.2	53.3	4.4
M	M	L	42.8	50.0	22.8	15.4	18.2	34.6	4.4
M	M	M	42.8	50.0	22.8	15.4	18.2	34.6	4.4
M	M	H	42.8	50.0	22.8	15.4	18.2	34.6	4.4
M	H	L	42.8	70.1	39.5	15.4	18.2	14.8	4.4
M	H	M	42.8	70.1	39.5	15.4	18.2	14.8	4.4
M	H	H	42.8	70.1	39.5	15.4	18.2	14.8	4.4
H	L	L	63.8	30.7	8.9	33.4	36.6	35.0	6.5
H	L	M	63.8	30.7	8.9	33.4	36.6	35.0	6.5
H	L	H	63.8	30.7	8.9	33.4	36.6	35.0	6.5
H	M	L	63.8	50.0	22.8	33.4	36.6	16.3	6.5
H	M	M	63.8	50.0	22.8	33.4	36.6	16.3	6.5
H	M	H	63.8	50.0	22.8	33.4	36.6	16.3	6.5
H	H	L	"N/A"	"N/A"	"N/A"	"N/A"	"N/A"	"N/A"	"N/A"
H	H	M	"N/A"	"N/A"	"N/A"	"N/A"	"N/A"	"N/A"	"N/A"
H	H	H	"N/A"	"N/A"	"N/A"	"N/A"	"N/A"	"N/A"	"N/A"

Note: The recall rate of the FDA's current practice in the testing set is 10.3%.

Appendix G: Robustness Check for Selection Bias

As noted in Section 2.1, our dataset, constructed from publicly available 510(k) records, includes only those devices that were granted market authorization by the FDA. Generally, there are two reasons a 510(k) submission does not proceed to clearance. The first is a Refusal to Accept (RTA), an administrative rejection based on application completeness. Because applicants may resubmit corrected applications, these devices typically appear in our dataset once clearance is eventually granted. The second is a Not Substantially Equivalent (NSE) determination, which may result either from predicate ineligibility (e.g., differences in intended use or technological characteristics) or from failure during substantive review (i.e., the device passed the eligibility check but failed performance testing).

The absence of NSE data does not affect model training, as recalls are post-market events and rejected devices therefore cannot be used for training our ML model. However, access to NSE determinations arising from failure during substantive review (as opposed to predicate ineligibility) could be informative for model evaluation. Internal analyses by the CDRH indicate that roughly 5% of all 510(k) submissions are ultimately classified as NSE (CDRH 2011). While this small fraction of rejections mitigates concerns about selection bias, it does not eliminate them. To address this limitation, we conduct a robustness check by augmenting our dataset with synthetic devices that simulate devices rejected by the FDA, and then re-evaluating our policy performance in terms of recall rate percentage improvement and rejection rate of safe devices.

In particular, we perform a sensitivity analysis by generating synthetic applicant devices rejected during the substantive review. To do this, we introduce a parameter ν that represents a risk quantile, which allows us to control the risk profile of synthetic devices. Given ν and the estimated recall risk from our chosen model, we first sample 5% of the original dataset from devices whose estimated risk exceeds $\nu \in \{0.60, 0.75, 0.95\}$. We then label these sampled devices as synthetic FDA-rejected cases. To reflect the assumption that FDA rejections for performance deficiencies correspond to actual safety risks, we assign a recall outcome to all synthetic devices. Consistent with our methodology, we do not use these synthetic devices to retrain our selected ML model, as real-world rejected devices lack the market outcomes required for training. Instead, we add these synthetic devices to the testing set and apply our existing policy, derived solely from cleared devices, to evaluate its performance. Consequently, FDA-rejected devices that are deferred by our policy for further evaluation by an FDA committee are assumed to end up receiving a rejection decision in our analysis.

Figure EC.4 depicts two key performance metrics in our representative setting (using the L-L-M combination), including the rejection rate of safe devices and the recall rate percentage improvement. As expected by the design of our analysis, we observe that the rejection rate of safe devices remains constant at the baseline level of 9.7%. Concurrently, the recall rate percentage improvement remains substantial, ranging from 30.8% to 23.9% across the different risk profiles. The improvement is highest (30.8%) when we simulate a targeted rejection of only the highest-risk devices ($\nu = 0.90$), which are easier for our policy to identify. As we broaden the scope to include lower-risk rejected devices ($\nu = 0.60$), the improvement decreases slightly to 23.9% but remains significant compared to the baseline FDA process. This robustness suggests that relying on cleared devices for model training, a structural necessity given the lack of market outcomes for rejected applications, does not undermine the general effectiveness of our proposed policy.

Appendix H: Post-Hoc Analysis

In this section, we delve deeper into assessing the performance of our proposed policy via post-hoc analyses. We start by examining the characteristics of the deferred medical devices based on some of the top predictors of recall risk. The results are summarized in Table EC.7. In this table, we also compare devices that are deferred by our proposed policy for more in-depth evaluation to those that have been correctly accepted or rejected. It is evident that the risk estimates are higher for unrecalled devices deferred by our policy compared to those that have been correctly accepted. Similarly, the risk estimates are lower for recalled devices deferred by the policy compared to those that have been correctly rejected. For unrecalled devices, deferred cases have modestly elevated recall histories and slightly newer predicates compared to accepted

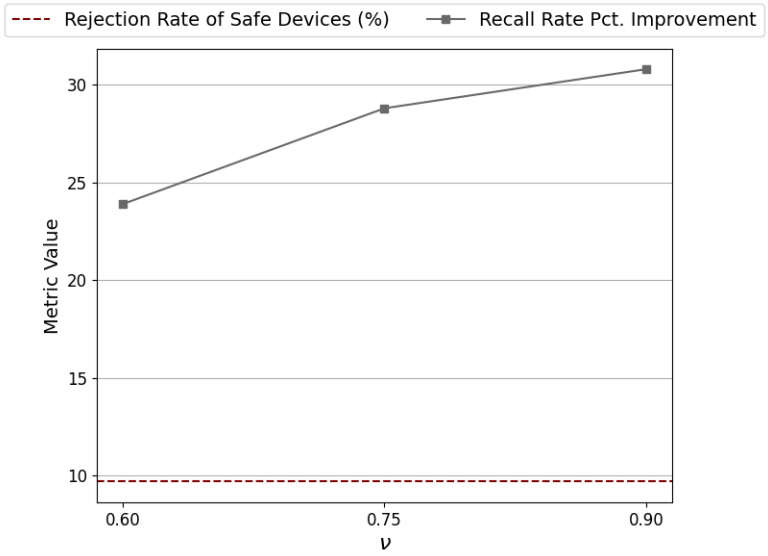

Figure EC.4 Impact of unobserved (FDA-rejected) devices on key metrics

Table EC.7 Comparison of devices correctly evaluated and deferred by our policy in the testing set

	Unrecalled Devices		Recalled Devices	
	Accepted	Deferred	Rejected	Deferred
	Mean (SD)	Mean (SD)	Mean (SD)	Mean (SD)
Risk Estimate	0.04 (0.01)	0.10 (0.03)	0.32 (0.14)	0.10 (0.03)
Predicate Median Age	4.74 (4.20)	3.81 (3.33)	3.49 (2.84)	3.86 (3.44)
Predicate Newest Age	4.57 (5.26)	3.63 (4.50)	2.84 (3.64)	3.46 (4.31)
Predicate Oldest Age	7.86 (6.70)	7.08 (6.32)	7.23 (6.55)	7.41 (6.86)
Num. of Class 2 Recalls	0.07 (0.56)	0.26 (1.00)	2.84 (5.04)	0.44 (2.99)
Weighted Num. of Recalls	0.02 (0.10)	0.09 (0.23)	0.57 (0.38)	0.11 (0.26)
Variance of Recalls	0.00 (0.05)	0.01 (0.10)	0.39 (2.87)	0.02 (0.22)

ones. For recalled devices, deferred cases exhibit lower recall histories and slightly older predicates than those that were confidently rejected. These patterns suggest that deferred devices fall into a borderline zone where risk signals and related features raise some concern, but not strongly enough to support a confident accept or reject decision. This supports the need for additional human review and the design of our policy to defer hard cases for further evaluation.

Table EC.8 presents a comparison of devices accepted by our proposed policy and those accepted based on the FDA's current practice in the testing set. As our policy does not directly diagnose the deferred devices, this analysis is conducted under the assumption that the deferred devices are evaluated following the FDA's current practice. We observe that the risk estimate for devices accepted by our proposed policy is relatively similar to that of the FDA's accepted devices (0.08 vs. 0.10). This indicates that both approaches accept devices with comparably low estimated risk; however, the predicate characteristics differ. The age of the latest-approved predicate for devices accepted under our policy is slightly higher compared to devices accepted under the FDA's current practice, while the age of the earliest-approved predicate is relatively

Table EC.8 Comparison of accepted devices by our policy and current practice in the testing set

	Our Policy	Current Practice
	Mean (SD)	Mean (SD)
Risk Estimate	0.08 (0.04)	0.10 (0.09)
Predicate Median Age	4.11 (3.66)	4.04 (3.58)
Predicate Newest Age	3.92 (4.76)	3.78 (4.65)
Predicate Oldest Age	7.35 (6.48)	7.33 (6.48)
Num. of Class 2 Recalls	0.21 (1.16)	0.49 (2.57)
Weighted Num. of Recalls	0.07 (0.20)	0.12 (0.28)
Variance of Recalls	0.01 (0.10)	0.06 (1.42)

the same. This shows a balance between leveraging proven safety records and embracing new technology. Furthermore, the number of Class 2 recalls, weighted number of recalls, and variance of predicates' recalls for devices accepted by our policy is significantly lower than those accepted by FDA's current practice. This is a substantial difference and indicates that our policy has the tendency to accept devices with predicates with fewer historical recalls, which aligns with the existing literature suggesting the correlation between the chance of recall and the number of recalls for predicates.

Appendix I: A Comparison of Human-Algorithm and ML-Only Policies

In this section, we compare the performance of our policy, which is a combined human-algorithm approach, with that of an ML-only policy that is solely based on non-human decisions. We conduct this comparison in two distinct parts: first, an analysis of operational trade-offs under a fixed set of preferences, and second, a more robust comparison of the policies' entire performance frontiers.

The recall rate percentage improvement relative to the FDA's current practice is an important metric. However, the overall rejection rate is also a critical operational factor for the FDA, as adopting a policy that results in a significant number of submission rejections may not be practical. Accordingly, we first compare the two policies based on these two metrics under a fixed set of preferences and a conservative evaluation. We solve our Primary Problem using the same set of parameters ($\lambda = 0.8$, $\xi^{ru} = 0.3$, and $\xi^{as} = 0.3$) while varying the threshold on the FDA's workload ρ from zero to one. The solution at $\rho = 0$ represents the optimal ML-only policy under these exact constraints, as a zero workload requirement forces the optimizer to find the best single-threshold solution. The solutions for $\rho > 0$ show the marginal benefit of the deferral option in our human-algorithm policy. This approach ensures both policies satisfy the same minimum performance constraints and use the same priority-weighting (λ).

Figure EC.5 shows the recall rate percentage improvement versus the overall rejection rate, with the FDA workload annotated by text on the curve. Under this fixed set of preferences, the ML-only policy ($\rho = 0$) yields a 90.5% improvement in recall rate, but does so by rejecting an impractical 71.6% of all submissions. Our policy navigates this trade-off more effectively. The red curve shows that our policy provides an operational lever, allowing a decision-maker to trade this high recall rate for a more balanced policy. For example, our policy can reduce the rejection rate to 12.1% and reduce the FDA's workload by 40.5%, while still achieving a

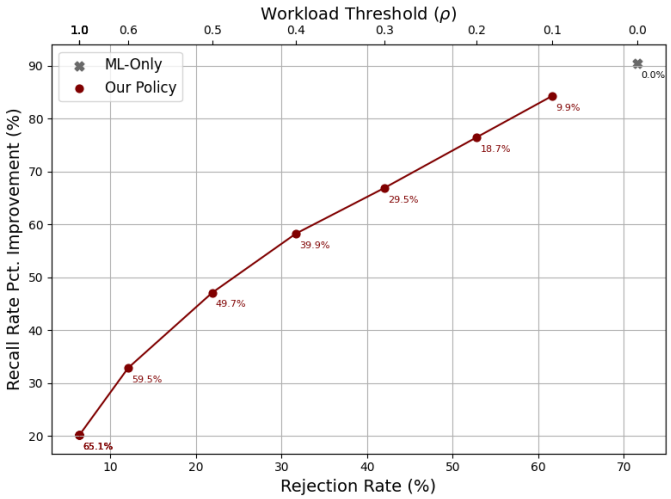

Figure EC.5 Comparison of the performance of our policy vs the ML-only policy in the testing set under ($\lambda = 0.8$, $\xi^{ru} = 0.3$, and $\xi^{as} = 0.3$). The annotated text on the curve indicates the corresponding FDA's workload.

32.9% improvement in recall rate. This demonstrates that even in a conservative setting, our policy provides critical flexibility.

While the first analysis shows that our policy provides a valuable operational trade-off, the other question is whether our policy is fundamentally superior compared to an ML-only policy. To answer this, we can compare the entire performance frontier of both policies by varying the priority-weighting parameter (λ). However, conducting this frontier-level comparison in the conservative setting is uninformative. Unlike the ML-only policy, which does not result in deferring any applicant devices, our policy results in deferring some proportion of applicant devices to the FDA committee for more in-depth evaluation in a setting with additional a priori information about the risk of the applicant device and lower workload. Consequently, the performance of our policy partly depends on the performance of the FDA committee to evaluate the deferred devices. In particular, for any risk thresholds (ℓ, h) selected by our policy, there exists a corresponding threshold for the ML-only policy by setting $t = h$. With this construction, both policies yield the same acceptance rate for unsafe devices and the same rejection rate for safe devices. Devices with estimated risk at or above h are rejected under both policies, as the ML-only threshold is the same as the rejection boundary of our policy. Devices with risk below ℓ are accepted by both approaches since the single threshold h is equal to or greater than ℓ. For devices with risk between ℓ and h, the ML-only policy accepts the device, whereas our policy defers the decision to the FDA. However, because our dataset contains only devices that were cleared (accepted) by the FDA, all deferred devices will be accepted as well.

Accordingly, we move to a non-conservative setting by modeling the added benefits of our policy. These benefits include providing additional information for deferred devices and reducing the FDA's overall workload. This can potentially allow the committee to better determine the review depth and level of scrutiny required for each applicant device; for example, riskier devices may undergo more rigorous evaluations, while the reduced workload allows committees to focus their expertise on the most complex cases. We assume our policy benefits the FDA by providing additional information and reducing workload, allowing the committee

to better evaluate deferred cases. We model the probability that the FDA's committees will fail to detect an unsafe deferred device as follows:

$$\mathcal{L}(f(X),k) = 2\left(1 - \frac{1}{1+\exp\left(-k(f(X)-\hat{\ell})\right)}\right),$$

where $f(X)$ is the estimated risk, $\hat{\ell}$ is the low threshold obtained by our policy, and parameter $k \geq 0$ is a scalar where higher values of k correspond to improved performance of the FDA's committees in detecting unsafe devices. When $k = 0$, this probability is equal to 1 by our design for any unsafe device that has been deferred. This implies that $k = 0$ is the baseline that matches the FDA's current practice in evaluating deferred devices without supplementary information.

Figure EC.6 illustrates the relationship between the predicted risk and the probability of failing to reject an unsafe device for 100 random samples, ordered by their predicted risk. As can be seen, the probability of failure decreases as the predicted risk of the deferred device increases. When $k = 0$, the FDA's committees will fail to reject an unsafe device with a probability of 1. As k increases, this probability decreases. This observation aligns with our intuition that unsafe devices appearing less risky are more challenging for the FDA's committees to detect. We acknowledge that this non-conservative analysis is stylized and relies on the specific parametric form of $\mathcal{L}(f(X),k)$ to model the potential benefits of deferral. Its purpose is to illustrate the potential for our policy to create a dominant frontier under improved expert performance, rather than to definitively model the complex operational trade-offs of time or resources.

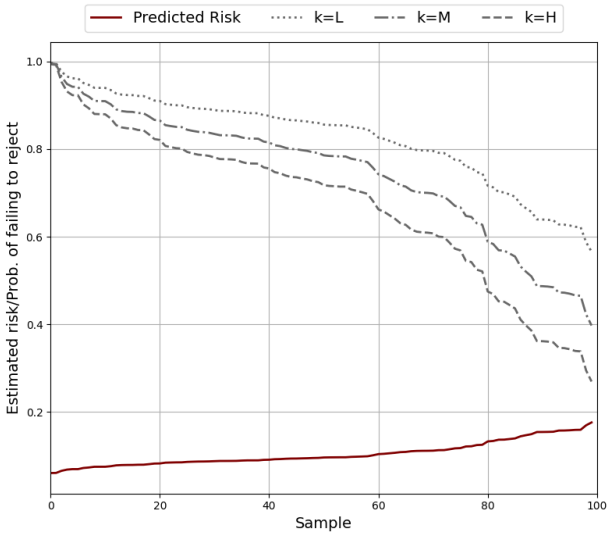

Figure EC.6 Predicted risk and probability of failing to reject unsafe deferred devices

Using this non-conservative framework, we trace the entire performance frontier for both policies by varying $\lambda \in [0.1, 0.9]$ while keeping the operational constraints fixed ($\xi^{ru} = 0.3$ and $\xi^{as} = 0.3$). Figure EC.7 shows the Pareto frontiers for three levels of FDA performance ($k \in \{L, M, H\}$), filtering for policies that reject fewer than 15% of safe devices to reflect more feasible cases in terms of implementation. As shown in Figure EC.7,

our policy's frontiers are strictly dominant over the ML-only policy across all levels of k. The performance gap is largest in the desirable operational region, where the rejection rate of safe devices is relatively low. We observe that the performance of our policy improves with higher levels of k. In other words, when the FDA committee becomes more effective at rejecting deferred unsafe devices, our policy's partial deferral strategy becomes increasingly beneficial. In practical terms, the ML-only policy is most effective when its risk estimations are highly accurate, where the ML-only policy can make accurate decisions for hard cases. However, in many real-world scenarios, including ours, risk estimation models perform reasonably well but lack uniform accuracy across all cases. In such settings, selectively deferring difficult cases to human experts can significantly enhance decision-making and lead to a better-performing policy overall.

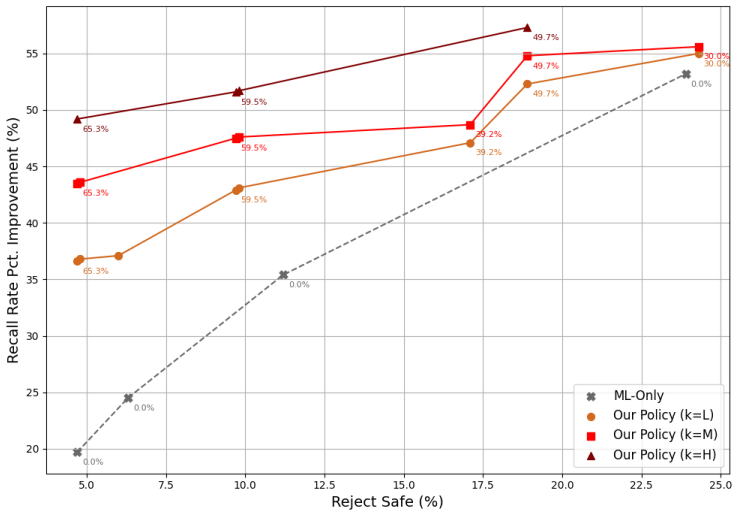

Figure EC.7 Comparison of the performance of our policy vs the ML-only policy in the testing set under ($\lambda \in [0.1, 0.9]$, $\xi^{ru} = 0.3$, and $\xi^{as} = 0.3$). The annotated text on each curve indicates the corresponding FDA's workload.

Appendix J: Estimation of Replacement Costs of Recalled Medical Devices

In this section, we discuss how the replacement costs of recalled medical devices are calculated. As noted by the OIG report, the replacement cost of recalled medical devices cannot be tracked to individual devices based solely on claims data, as Medicare claim forms do not contain a field for reporting medical device-specific information (HHS 2017). Consequently, our estimate of replacement costs is specific to the medical specialty, not individual devices. We determined the medical specialty for over 99% of the 1,351 unique HCPCS codes/descriptions in the MDMEDS 2013-2020 data by first identifying keywords in the device classification names available in the 510(k) submission data for each medical specialty and then matching those keywords to the HCPCS descriptions in the MDMEDS data. For example, an HCPCS description containing the word "ostomy," which is a procedure used to treat various diseases of the urinary or digestive systems, was classified as having the medical specialty Gastroenterology/Urology. In a slightly more intricate case, HCPCS descriptions containing the words "glucose monitor" or "glucose" and "monitor" were classified

as having the medical specialty Clinical Chemistry. Table EC.9 shows the crosswalk between keywords and medical specialties.

Table EC.9 Medical Specialties and Keywords

Medical Specialty	**Keyword(s)**
Anesthesiology	"aerosol" and "compressor", "airway", "breathing circuits", "cough", "face mask", "nasal cannula", "nasal mask", "nebulization", "nebulizer", "oropharyngeal", "oxygen", "positive expiratory pressure", "respiratory", "tracheal suction", "ventilator"
Cardiovascular	"defibrillator", "pneumatic compression device"
Clinical Chemistry	"calibrator solution", "glucose monitor", "glucose" and "monitor"
Dental	"osteogenesis"
Gastroenterology/Urology	"bladder", "cervical", "drainage bag", "indwelling catheter", "insertion tray" and "catheter", "leg strap", "male" and "catheter", "ostomy", "parenteral", "pelvic floor", "stoma cap", "urethral", "urinary"
General & Plastic Surgery	"adhesive", "bandage", "chest wall", "collagen" and "wound", "compression" and "wrap", "dressing", "gauze", "lancet", "skin barrier", "sterile water", "tape", "tubing" and "pump", "ultraviolet" and "therapy", "wound"
General Hospital	"ambulatory infusion pump", "bath", "bed", "canister" and "pump", "chair", "compression stocking", "compression" and "garment", "compressor" and "for equipment", "drug infusion", "footplate", "footrests", "heel loop", "infusion pump", "insulin", "irrigation", "iv pole", "lubricant", "mattress", "transfer device", "urinal" and "jug-type"
Neurology	"conductive garment", "nerve stimulation"
Physical Medicine	"armrest", "cane", "commode chair", "crutches", "electrical" and "stimulator", "flexion", "foot" and "density insert", "foot" and "shoe molded", "heat pad", "inlay" and "shoe", "knee" and "exercise", "leg" and "compressor", "neuromuscular stimulator", "patient lift", "patient support system", "patient transfer", "pneumatic" and "compressor", "rear wheel", "traction" and "cervical", "trapeze", "vehicle", "walker", "wheelchair"

Once the medical specialties for the HCPCS codes were established, we calculated the average Medicare allowed amount per medical specialty. This calculation was based on the total supplier claims, which reflect the number of products ordered by the referring provider. The Medicare allowed amount includes the amount Medicare paid, the deductible and coinsurance amounts owed by the beneficiary, as well as any amount owed by a third-party payer (CMS 2021). We were able to calculate the average Medicare allowed amount for over 75% of the devices in our test dataset based on their respective medical specialties. However, for the devices for which we were unable to compute the average Medicare allowed amount, we faced a challenge in assigning a specific medical specialty to them. Among them, Radiology devices account for 13.7% of all devices in the test dataset and the remaining specialties cumulatively account for less than 8.5% of devices. We believe that the chief reason we could not calculate Medicare costs for these specialties is that the underlying 510(k) devices are not single-use devices, expended on a single patient. In the case of radiology, we examined the majority of 510(k) cleared products between 2013 and 2020 and determined that these devices were either imaging software or multi-use equipment used in providing radiology services such as radiosurgery or radiotherapy. For the purposes of our impact assessment, we created low and high estimates

of the average Medicare allowed amount when we could not directly calculate the average Medicare allowed amount. The low estimate is the lowest average Medicare allowed amount across all of the specialties. The high estimate is the weighted average Medicare allowed amount across all specialties, with weights derived from the frequency of each medical specialty in our testing dataset.

Appendix K: Proofs

All proofs for lemmas, propositions, and theorems are given below.

PROPOSITION 1. *For any $\theta \in (0,1)$ and a pair of $h(\xi^{ru})$ and $\ell(\xi^{as})$ in the Auxiliary Problem, we have:*

(a) *if $h(\xi^{ru}) > \ell(\xi^{as})$, then $(\ell(\xi^{as}), h(\xi^{ru}))$ is the unique optimal solution,*

(b) *if $h(\xi^{ru}) < \ell(\xi^{as})$, then the problem is infeasible,*

(c) *if $h(\xi^{ru}) = \ell(\xi^{as})$, then $\ell = h = h(\xi^{ru}) = \ell(\xi^{as})$ is the single threshold optimal solution.*

Proof of Proposition 1: We prove each case separately.

Case (a). We first find an optimal solution, and then we prove its uniqueness. When $h(\xi^{ru}) > \ell(\xi^{as})$, the polyhedral feasible region of the problem has three extreme points: $(\ell(\xi^{as}), \ell(\xi^{as}))$, $(\ell(\xi^{as}), h(\xi^{ru}))$, and $(h(\xi^{ru}), h(\xi^{ru}))$. The objective values corresponding to these extreme points are $\ell(\xi^{as})(1-2\theta)$, $-\theta\,\ell(\xi^{as}) + (1-\theta)h(\xi^{ru})$, and $h(\xi^{ru})(1-2\theta)$, respectively. For any $\theta \in (0,1)$, we have $-\theta\,\ell(\xi^{as}) + (1-\theta)h(\xi^{ru}) > \ell(\xi^{as})(1-2\theta)$ because $h(\xi^{ru}) > \ell(\xi^{as})$ and $\theta < 1$. Also, for any $\theta \in (0,1)$, we have $-\theta\,\ell(\xi^{as}) + (1-\theta)h(\xi^{ru}) > h(\xi^{ru})(1-2\theta)$ because $h(\xi^{ru}) > \ell(\xi^{as})$ and $\theta > 0$. Accordingly, $(\ell(\xi^{as}), h(\xi^{ru}))$ is an optimal solution for the problem.

Next, we use a contradiction argument to prove that $(\ell(\xi^{as}), h(\xi^{ru}))$ is the unique optimal solution. Suppose that there is another optimal solution $(\bar{\ell}, \bar{h}) \neq (\ell(\xi^{as}), h(\xi^{ru}))$. According to the polyhedral feasible region, there are two possible scenarios: (1) $\bar{h} < h(\xi^{ru})$ and $\bar{\ell} \geq \ell(\xi^{as})$, or (2) $\bar{h} \leq h(\xi^{ru})$ and $\bar{\ell} > \ell(\xi^{as})$. In both scenarios, we have:

$$-\theta\,\bar{\ell} + (1-\theta)\,\bar{h} < -\theta\,\ell(\xi^{as}) + (1-\theta)\,h(\xi^{ru}).$$

This contradicts the optimality of $(\bar{\ell}, \bar{h})$. Thus, we conclude that $(\ell(\xi^{as}), h(\xi^{ru}))$ is the unique optimal solution.

Case (b). The condition of $h(\xi^{ru}) < \ell(\xi^{as})$, results in $h < \ell$, which contradicts the requirement of $0 \leq \ell \leq h \leq 1$. Thus, the problem is infeasible.

Case (c). This is a direct result of Case (a).

Q.E.D.

THEOREM 1. *For any $\theta \in (0,1)$ and $\lambda \in (0,1)$, and a pair of $h(\xi^{ru})$ and $\ell(\xi^{as})$, we have:*

(a) *if $h(\xi^{ru}) > \ell(\xi^{as})$, then the two-threshold optimal solution of the Auxiliary Problem is optimal in the Relaxed Problem,*

(b) *if $h(\xi^{ru}) < \ell(\xi^{as})$, then both problems are infeasible,*

(c) *if $h(\xi^{ru}) = \ell(\xi^{as})$, then the single threshold optimal solution of the Auxiliary Problem is optimal in the Relaxed Problem.*

Proof of Theorem 1: First, we highlight that both problems have the same polyhedral feasible region. Note that $h, \ell \in [0,1]$ and the interval $[0,1]$ is convex and compact. Supremum is attained since $\mathbb{P}(\delta^+ \geq h)$ is continuous and weakly decreasing in h. Similarly, infimum is attained since $\mathbb{P}(\delta^- \leq \ell)$ is continuous and weakly increasing in ℓ. Accordingly, we have:

$$\mathbb{P}(\delta^+ \geq h) \geq \xi^{ru} \iff h \leq h(\xi^{ru}), \text{ and } \quad \mathbb{P}(\delta^- \leq \ell) \geq \xi^{as} \iff \ell \geq \ell(\xi^{as}).$$

Hence, any feasible solution of the Auxiliary Problem is also a feasible solution to the Relaxed Problem.

Next, we show that both problems have the same optimal solutions in each case.

Case (a). By Proposition 1, when $h(\xi^{ru}) > \ell(\xi^{as})$, we have that $(\ell(\xi^{as}), h(\xi^{ru}))$ is the unique optimal solution of the Auxiliary Problem for any $\theta \in (0,1)$. The objective function of the Relaxed Problem is the convex combination of $\mathbb{P}(\delta^+ \leq \ell)$ and $\mathbb{P}(\delta^- \geq h)$, which are monotone functions. Since $\mathbb{P}(\delta^+ \leq \ell)$ is weakly increasing in ℓ, we have $\mathbb{P}(\delta^+ \leq \ell) \geq \mathbb{P}(\delta^+ \leq \ell(\xi^{as}))$ for any $\ell \geq \ell(\xi^{as})$. Similarly, since $\mathbb{P}(\delta^- \geq h)$ is weakly decreasing in h, we have $\mathbb{P}(\delta^- \geq h) \geq \mathbb{P}(\delta^- \geq h(\xi^{ru}))$ for any $h \leq h(\xi^{ru})$. Accordingly, for any feasible pair of ℓ and h, we have:

$$\lambda\, \mathbb{P}(\delta^+ \leq \ell) + (1-\lambda)\, \mathbb{P}(\delta^- \geq h) \geq \lambda\, \mathbb{P}(\delta^+ \leq \ell(\xi^{as})) + (1-\lambda)\, \mathbb{P}(\delta^- \geq h(\xi^{ru})).$$

Therefore, we can conclude that $(\ell(\xi^{as}), h(\xi^{ru}))$ is also the optimal solution of the Relaxed Problem.

Case (b). By Proposition 1, when $h(\xi^{ru}) < \ell(\xi^{as})$, the Auxiliary Problem is infeasible. The Relaxed Problem is also infeasible since both problems are restricted to the same set of constraints.

Case (c). This is a direct result of Case (a).

Q.E.D.

THEOREM 2. *For the Primary Problem, we have:*

(a) *If $h(\xi^{ru}) > \ell(\xi^{as})$ and the workload constraint (3) holds for $\ell = \ell(\xi^{as})$ and $h = h(\xi^{ru})$, then $(\ell(\xi^{as}), h(\xi^{ru}))$ is an optimal solution,*

(b) *If $h(\xi^{ru}) > \ell(\xi^{as})$ and the workload constraint (3) does not hold for $\ell = \ell(\xi^{as})$ and $h = h(\xi^{ru})$, then for any given $\epsilon > 0$, our nested search algorithm returns an ϵ-optimal solution after at most*

$$\mathcal{O}\left(\frac{\bar{L}\,(h(\xi^{ru}) - \ell(\xi^{as}))}{\epsilon} \log_2\left(\frac{2\bar{L}_\phi(1-\lambda)\,(h(\xi^{ru}) - \ell(\xi^{as}))}{\epsilon}\right)\right)$$

iterations, where $\lambda \in (0,1)$, $\bar{L}_\phi$ and $\bar{L}_h$ are, respectively, upper bounds on the common Lipschitz constant of $\phi^{\pm}(\cdot)$ and the Lipschitz constant of $h^*(\ell)$, and $\bar{L} = \bar{L}_\phi\left(\lambda + (1-\lambda)\bar{L}_h\right)$. Setting $\Delta = \epsilon/[2\bar{L}]$ and $\zeta = \epsilon/[2(1-\lambda)\bar{L}_\phi]$ guarantees ϵ-optimality and ensures that the workload constraint is satisfied within a tolerance of $\epsilon/[2(1-\lambda)]$,

(c) *If $h(\xi^{ru}) < \ell(\xi^{as})$, then the problem is infeasible,*

(d) *If $h(\xi^{ru}) = \ell(\xi^{as})$, then the problem has a single threshold solution $\ell^* = h^* = h(\xi^{ru}) = \ell(\xi^{as})$.*

Proof of Theorem 2: We prove each case separately.

Case (a). Since $(\ell(\xi^{as}), h(\xi^{ru}))$ meets the FDA's workload constraint (3), the Primary Problem becomes equivalent to the Relaxed Problem. By Theorem 1, if $h(\xi^{ru}) > \ell(\xi^{as})$, then the optimal solution of the Auxiliary Problem is optimal in the Relaxed Problem. Consequently, $(\ell(\xi^{as}), h(\xi^{ru}))$ is also an optimal solution of the Primary Problem.

Case (b). In this case, $(\ell(\xi^{as}), h(\xi^{ru}))$ is not a feasible solution because it violates the FDA's workload constraint by assumption. In particular, the problem has the following feasible region:

$$\Theta = \Big\{(\ell, h) \text{ s.t. } q\left(\phi^+(h) - \phi^+(\ell)\right) + (1-q)\left(\phi^-(h) - \phi^-(\ell)\right) \le \rho, h \le h(\xi^{ru}), \ell \ge \ell(\xi^{as})\Big\}.$$

We show that our nested search finds a near-optimal solution within a finite number of steps. To do so, we bound the sub-optimality gap $J(\ell^{NS}, h^{NS}) - J(\ell^*, h^*)$, where $J(\cdot)$ is the objective function of the Primary Problem.

Recall that

$$g(\ell, h) = q\big(\phi^+(h) - \phi^+(\ell)\big) + (1-q)\big(\phi^-(h) - \phi^-(\ell)\big) - \rho.$$

Under the conditions of Case (b), there exists an optimal solution at which the workload constraint binds, so that $g(\ell^*, h^*) = 0$.

Let $\ell_k = \max\{\ell \in \mathcal{L} : \ell \le \ell^*\}$, that is, ℓ_k is the grid point immediately to the left of ℓ^*. Since the grid has step size Δ, we have $0 \le \ell^* - \ell_k \le \Delta$.

We next show that $\ell_k \in \mathcal{L}_{\text{feas}}$. Since $g(\ell, h)$ is nonincreasing in ℓ and nondecreasing in h, and since $\ell_k \le \ell^*$ and $h^* \le h(\xi^{ru})$, we have

$$g(\ell_k, h(\xi^{ru})) \ge g(\ell^*, h(\xi^{ru})) \ge g(\ell^*, h^*) = 0.$$

Moreover, we have $g(\ell_k, \ell_k) = -\rho \le 0$. Therefore, there exists $h^*(\ell_k) \in [\ell_k, h(\xi^{ru})]$ such that $g(\ell_k, h^*(\ell_k)) = 0$. Hence, $\ell_k \in \mathcal{L}_{\text{feas}}$.

By design of our algorithm, ℓ^{NS} is selected such that it yields a value less than or equal to the value at any other feasible grid point. Therefore, $J(\ell^{NS}, h^{NS}) \le J(\ell_k, h^*_\zeta(\ell_k))$.

Accordingly, we decompose the optimality gap as follows:

$$\begin{aligned} J(\ell^{NS}, h^{NS}) - J(\ell^*, h^*) &\le J(\ell_k, h^*_\zeta(\ell_k)) - J(\ell^*, h^*) \\ &= \underbrace{\left[J(\ell_k, h^*_\zeta(\ell_k)) - J(\ell_k, h^*(\ell_k))\right]}_{\text{Sub-optimality due to Bisection Search}} + \underbrace{\left[J(\ell_k, h^*(\ell_k)) - J(\ell^*, h^*)\right]}_{\text{Sub-optimality due to Grid Search}}, \end{aligned}$$

where $h^*(\ell)$ is the exact root of equation (4) for a feasible grid point $\ell \in \mathcal{L}_{\text{feas}}$.

We proceed with the proof in two main steps.

Step 1 (Sub-optimality due to Bisection Search for h). The first term can be expanded as:

$$J(\ell_k, h^*_\zeta(\ell_k)) - J(\ell_k, h^*(\ell_k)) = (1-\lambda)\left(\phi^-(h^*(\ell_k)) - \phi^-(h^*_\zeta(\ell_k))\right).$$

Assuming $\phi^-(\cdot)$ is L_ϕ-Lipschitz, the above term can be upper bounded as follows:

$$J(\ell_k, h^*_\zeta(\ell_k)) - J(\ell_k, h^*(\ell_k)) \le (1-\lambda) L_\phi \left|h^*(\ell_k) - h^*_\zeta(\ell_k)\right|.$$

Let $h^{(n)}(\ell)$ denote the solution of the bisection after n iterations for $\ell \in \mathcal{L}_{\text{feas}}$. The bisection search generates a sequence of $\{h^{(n)}(\ell)\}_{n=1}^{\infty}$ approximating the root of the workload constraint such that (Gautschi 2011):

$$\left|h^{(n)}(\ell) - h^*(\ell)\right| \le \frac{h(\xi^{ru}) - \ell(\xi^{as})}{2^n}, \qquad \forall n \ge 1.$$

Selecting the tolerance as $\zeta = \frac{h(\xi^{ru})-\ell(\xi^{as})}{2^n}$, the number of iterations to reach this tolerance is at most $\log_2\left(\frac{h(\xi^{ru})-\ell(\xi^{as})}{\zeta}\right)$. Thus, the first term of the sub-optimality gap is at most $(1-\lambda)L_\phi\zeta$ after $\log_2\left(\frac{h(\xi^{ru})-\ell(\xi^{as})}{\zeta}\right)$ iterations.

Next, we show that the solution of the bisection search, h^{NS}, satisfies the workload constraint within a tolerance level. Recall that for a given ℓ, we have

$$g(\ell, h) = q\big(\phi^+(h) - \phi^+(\ell)\big) + (1-q)\big(\phi^-(h) - \phi^-(\ell)\big) - \rho.$$

Also, by definition, $g(\ell^{NS}, h^*(\ell^{NS})) = 0$. Therefore,

$$\begin{aligned}
|g(\ell^{NS}, h^{NS})| &= \left|g(\ell^{NS}, h^{NS}) - g(\ell^{NS}, h^*(\ell^{NS}))\right| \\
&= \left|q\big(\phi^+(h^{NS}) - \phi^+(h^*(\ell^{NS}))\big) + (1-q)\big(\phi^-(h^{NS}) - \phi^-(h^*(\ell^{NS}))\big)\right| \\
&\le q\left|\phi^+(h^{NS}) - \phi^+(h^*(\ell^{NS}))\right| + (1-q)\left|\phi^-(h^{NS}) - \phi^-(h^*(\ell^{NS}))\right| \\
&\le qL_\phi\left|h^{NS} - h^*(\ell^{NS})\right| + (1-q)L_\phi\left|h^{NS} - h^*(\ell^{NS})\right| \\
&\le qL_\phi\zeta + (1-q)L_\phi\zeta \\
&= L_\phi\zeta.
\end{aligned}$$

Step 2 (Sub-optimality due to Grid Search for ℓ). Let $F^*(\ell) = \lambda\phi^+(\ell) + (1-\lambda)\left(1-\phi^-(h^*(\ell))\right)$. By this definition, we have:

$$J(\ell_k, h^*(\ell_k)) - J(\ell^*, h^*) = F^*(\ell_k) - F^*(\ell^*).$$

We now establish the Lipschitz constant for $F^*(\ell)$. We assume $\phi^+(\cdot)$ is L_ϕ-Lipschitz. Then the first term, $\lambda\phi^+(\ell)$, is (λL_ϕ)-Lipschitz by the multiplication property. For the second term, by assumption, $\phi^-(\cdot)$ is L_ϕ-Lipschitz and $h^*(\ell)$ is L_h-Lipschitz. By the composition property, the composite function $1-\phi^-(h^*(\ell))$ is $(L_\phi L_h)$-Lipschitz. It follows from the scalar multiplication property that the entire second term is $((1-\lambda)L_\phi L_h)$-Lipschitz. By the sum property, the Lipschitz constant for $F^*(\ell)$ is the sum of the constants of its terms: $\lambda L_\phi + (1-\lambda)L_\phi L_h = L$.

Accordingly, we can upper bound the sub-optimality due to the grid search as follows:

$$J(\ell_k, h^*(\ell_k)) - J(\ell^*, h^*) \le L|\ell_k - \ell^*| \le L\Delta,$$

where the inequality holds by the Lipschitz property of $F^*(\cdot)$ and the fact that $0 \le \ell^* - \ell_k \le \Delta$.

As a result, our nested search yields an ϵ-optimal solution (ℓ^{NS}, h^{NS}) after at most $\mathcal{O}\left(\frac{h(\xi^{ru})-\ell(\xi^{as})}{\Delta}\log_2\left(\frac{h(\xi^{ru})-\ell(\xi^{as})}{\zeta}\right)\right)$ iterations. The workload feasibility is satisfied within $L_\phi\zeta$, and the sub-optimality is bounded by $\epsilon = L\Delta + (1-\lambda)L_\phi\zeta$.

Next, we compute the maximum number of iterations required for any given $\epsilon > 0$. While L_ϕ and L_h are generally unknown, we can use known upper bounds, $\bar{L}_\phi \geq L_\phi$ and $\bar{L}_h \geq L_h$. Therefore, for a given value of ϵ, we can run our nested search algorithm with

$$\Delta = \frac{\epsilon}{2\bar{L}}, \qquad \text{and} \qquad \zeta = \frac{\epsilon}{2(1-\lambda)\bar{L}_\phi}.$$

Under these settings, the sub-optimality is:

$$J(\ell^{NS}, h^{NS}) - J(\ell^*, h^*) \leq L\Delta + (1-\lambda)L_\phi\zeta = \frac{L}{\bar{L}}\frac{\epsilon}{2} + \frac{L_\phi}{\bar{L}_\phi}\frac{\epsilon}{2} \leq \epsilon.$$

Similarly, the workload feasibility is satisfied with the desired tolerance:

$$|g(\ell^{NS}, h^{NS})| \leq L_\phi\zeta = \frac{L_\phi}{\bar{L}_\phi}\frac{\epsilon}{2(1-\lambda)} \leq \frac{\epsilon}{2(1-\lambda)}.$$

Accordingly, our nested search returns an ϵ-optimal solution while keeping the workload feasibility within $\epsilon/[2(1-\lambda)]$. Substituting the values of Δ and ζ, the maximum number of iterations required is:

$$\mathcal{O}\left(\frac{\bar{L}\left(h(\xi^{ru}) - \ell(\xi^{as})\right)}{\epsilon}\log_2\left(\frac{2(1-\lambda)\bar{L}_\phi\left(h(\xi^{ru}) - \ell(\xi^{as})\right)}{\epsilon}\right)\right).$$

Case (c). The condition of $h(\xi^{ru}) < \ell(\xi^{as})$ results in $h < \ell$, which contradicts the requirement of $0 \leq \ell \leq h \leq 1$ in the Primary Problem. Thus, the problem is infeasible.

Case (d). This can be shown by following a closely analogous argument to Case (a).

Q.E.D.

Appendix L: Nested Search Algorithm

This section presents the pseudocode of Algorithm 1 for the nested search procedure discussed in Section 4.3. This algorithm is employed to determine the risk thresholds (ℓ, h) when the workload constraint is binding.

Algorithm 1 Nested Search Algorithm

Input: Empirical CDFs ϕ^+ and ϕ^-; parameters q, λ, ρ, ξ^{as}, ξ^{ru}; grid step Δ; tolerance ζ
Initialize: Set of feasible candidates $\mathcal{L}_{\text{feas}} \leftarrow \emptyset$
Construct a grid $\mathcal{L}$ over $[\ell(\xi^{as}), h(\xi^{ru}))$ using the grid step Δ
for each $\ell \in \mathcal{L}$ **do**
    Set $h_L \leftarrow \ell$, $h_R \leftarrow h(\xi^{ru})$
    Compute $g(\ell, h_L)$ and $g(\ell, h_R)$, where $g(\ell, h) = q(\phi^+(h) - \phi^+(\ell)) + (1-q)(\phi^-(h) - \phi^-(\ell)) - \rho$
    if $g(\ell, h_L) > 0$ or $g(\ell, h_R) < 0$ **then**
        continue to next ℓ
    while $|h_R - h_L| > \zeta$ **do**
        $h_{\text{mid}} \leftarrow (h_L + h_R)/2$
        Compute $g(\ell, h_{\text{mid}})$
        if $g(\ell, h_{\text{mid}}) < 0$ **then**
            $h_L \leftarrow h_{\text{mid}}$
        else
            $h_R \leftarrow h_{\text{mid}}$
    Set $h^*_\zeta(\ell) \leftarrow (h_L + h_R)/2$
    Add ℓ to the set of feasible candidates $\mathcal{L}_{\text{feas}}$
    Compute $F(\ell) = \lambda\phi^+(\ell) + (1-\lambda)(1 - \phi^-(h^*_\zeta(\ell)))$ and record it
Select the thresholds as:

$$\ell^{NS} = \arg\min_{\ell \in \mathcal{L}_{\text{feas}}} F(\ell), \quad h^{NS} = h^*_\zeta(\ell^{NS})$$

References

Gautschi W (2011) *Numerical Analysis* (Boston: Birkhäuser).

Wasserman L, Roeder K (2009) High dimensional variable selection. *Annals of Statistics* 37(5A):2178.

Zuckerman D, Brown P, Das A (2014) Lack of publicly available scientific evidence on the safety and effectiveness of implanted medical devices. *JAMA Internal Medicine* 174(11):1781–1787.