Weather Index Insurance
Education Tool User Guide
Weather Index Insurance Education Tool User Guide
# Table of Contents

1. Introduction .................................................................................................................. 1  
2. System Requirements .................................................................................................... 2  
   Browser Requirements .................................................................................................... 2  
   Internet Explorer ........................................................................................................... 2  
   Firefox ........................................................................................................................... 2  
   Safari ............................................................................................................................ 3  
3. WIET Website .................................................................................................................. 4  
   Site Navigation ............................................................................................................... 4  
   Main Menu .................................................................................................................... 5  
   Crop Water Satisfaction Analysis Module ....................................................................... 5  
      Prerequisites ............................................................................................................. 5  
      Running Crop Water Stress Accounting .................................................................. 5  
      Results ...................................................................................................................... 7  
   Loss Conversion Module ................................................................................................ 7  
      Results ...................................................................................................................... 8  
   Create Contract Module ................................................................................................. 8  
      Results ...................................................................................................................... 9  
   Pricing Module ............................................................................................................... 10  
      Results .................................................................................................................... 11  
   Tune Contract ................................................................................................................ 11  
      Illustrative Example ............................................................................................... 13  
      Results .................................................................................................................... 13  
   Contract Evaluation ....................................................................................................... 13  
      Results .................................................................................................................... 14  
   Rainfall Simulator .......................................................................................................... 14  
4. Managing Data ............................................................................................................... 16  
   Manage Data .................................................................................................................. 16  
      Download or Delete Data ......................................................................................... 16  
      Uploading Data ......................................................................................................... 16  
   Download Data from Data Library .................................................................................. 19  
   Classrooms .................................................................................................................... 19  
      Create a Classroom .................................................................................................... 19  
      View/Modify a Classroom ......................................................................................... 19
List of Tables

4.1. Precipitation Data (Excel Table) .................................................................................. 17
4.2. Precipitation Data: File Format ................................................................................... 17
List of Examples

3.1. For Example ................................................................. 10
3.2. Example ................................................................. 13
4.1. Excel Example ............................................................. 17
Chapter 1. Introduction

Index insurance is a relatively new weather risk management tool. While traditional insurance insures against crop failure, index insurance insures for a specific event or risk, such as rainfall deficits. The index insurance can be more cost effective since there is no need for in-field assessment of damage because payouts are triggered by weather data directly. Index insurance addresses two problems associated with traditional crop insurance: moral hazard (incentives for a farmer to let a crop die in order to get an insurance payout) and adverse selection (in which insurance is priced based on the risks of the entire population but only the most vulnerable farmers purchase insurance).

Although index insurance may be able to address many of the problems with more traditional insurance, index insurance has its own limitations. Perhaps most importantly, index insurance is vulnerable to basis risk. Simply put, basis risk is when insurance payouts do not match actual losses – either there are losses but no payout, or a payout is triggered even though there are no losses. Obviously, if either of these situations occurs too frequently, the insurance scheme will not be viable, and may even damage livelihoods. The contract design, and in particular the selection of an appropriate index, is crucially important in minimizing basis risk.

This tool is intended to help educate people on the design of an insurance index using the example of a rainfall index. Through use of the tool, the you can design, evaluate, and compare hypothetical insurance indexes, becoming familiar with some common techniques in index design. You can also learn about important trade-offs as well as ways to compare strengths and weaknesses of different index design choices.


Download the Rainfall Simulation Technical Appendix: AppendixA.pdf
Chapter 2. System Requirements

Browser Requirements

WIIET supports the use of Internet Explorer v7.0 or higher, Firefox v3.0 or higher and Safari v3.5 or higher. Due to the extensive use of Javascript, older browsers may not work properly or display contents incorrectly.

When using your browser, make sure the browser settings allow Popup windows and Javascript.

Internet Explorer

Open the Tools->Internet Options from the browser

Make sure that the popup blocker is disabled.

Firefox

Open the Firefox Options/Preferences and set select
Make sure that **Block Popup Windows** is turned off (or set an exception for the portal.iri.columbia.edu website), and insure that **Enable Javascript** is enabled.

**Safari**

Open the Safari Preferences and select "Security"

Make sure that **Block Popups** is turned off, and that **Enable Javascript** is turned on.
Chapter 3. WIIET Website

WIIET stands for the Weather Index Insurance Educational Tool and is intended to teach the contract design process for index-based insurance contracts by allowing the user to create a contract by working through a series of modules. In this way, the user is able to walk through the contract design process step by step: from the collection of crop and climate information to selecting the contract parameters; from tuning those contract parameters to eventually evaluating the contract’s performance. The end result is to have a contract that successfully models and hedges against the climate risk that farmers face in real-world conditions and that offers protection at a price which is not prohibitively expensive.

As it is intended for educational purposes, the Online Educational Tool operates using a “classroom” system. A “classroom” is simply a learning environment that allows a teacher to select the specific data sets available to a class of students during the sessions allowing each class to have a custom experience.

The central challenge in index insurance is to provide a contract that effectively provides payouts when crop yields are bad. To perform this task, it is necessary to have some information on when the insured crops are doing badly. To allow this, the tool includes the Crop Water Satisfaction Analysis module that generates annual data on crop behavior based on a simple model using historical rainfall. The user then utilizes the Loss Conversion module to convert crop productivity to a dataset that roughly estimates financial losses due to drought.

In the Create Contract module, you can create a contract and apply it to a historical rainfall dataset to see what payouts would have occurred based on the rainfall. The Pricing module allows you to calculate an insurance premium that is based on the payouts calculated in the Create Contract module.

Other modules allow you to link the insurance to the losses. In the Tune Contract module, the tool will take a contract and attempt to adjust it to better reduce the risk represented in the loss datasets you have generated. The Contract Evaluation module provides a set of diagnostics and comparisons allowing you to evaluate how well a set of index insurance payouts protects against the risks in a loss dataset. With these two tools you can explore how to improve an index to provide as much protection as possible.

These comparisons using historical rainfall, known as ‘historical burn analysis’, can be expanded using the Rainfall Simulation module. This module uses an existing dataset to generate hundreds of years of hypothetical rainfall that has the same statistical properties as the historical rainfall. You can repeat your analysis using this data to include events that are possible, but that may not have occurred, in the past.

Site Navigation

WIIET is divided into a series of modules, each of which has a specific educational purpose. These modules can be accessed through the tab structure at the top of the page. The modules represent one stage of the design process but are not necessarily intended to be completed in chronological order.

The modules are designed to have two separate screens: one where the parameters are selected for analysis, and one where the results of the computation are shown. You may switch between the input page and the results page by clicking on the banner heading for each one. In this manner, it is easier to fine-tune the parameters to arrive at the results desired.
Many of the modules utilize pop-up windows for results and dialog, so you must make sure that pop-up windows are allowed in your browser.

As mentioned, the modules presented in WIIET are not necessarily intended to be completed in a strict order. Because robust contracts will often require many attempts until the right combination of parameters is released, the modules may need to be reiterated multiple times in sequence. WIIET supports this non-linear flow by allowing the user to save the results from each module for use in later modules.

For this reason, the modules fall into three broad categories:

- **Crop Productivity modules** where you will model the farmers’ historical losses based on the rainfall dataset and the amount of rainfall that the crops require.
- **Contract Creation modules** where you will create a contract.
- **Tools For Further Analysis** which contains some advanced metrics for evaluating your contract’s performance.

### Main Menu

The Main Menu is the first page you see when entering the Online Educational Tool. It provides you with links to each of the modules, as well as to the Manage Data screen.

You may also change the display language using the drop-down menu on the top right of the screen.

Use the links provided on the screen to begin any of these tasks.

### Crop Water Satisfaction Analysis Module

The Crop Water Satisfaction Module is a simple model of crop production based on historical rainfall data and average potential evapotranspiration data. The rainfall dataset is daily rainfall over several years, while the evapotranspiration data is the average evapotranspiration over all years. Agrometerological data for a crop is used to calculate how well the crop’s water demands are satisfied by the rainfall. This model is similar to the Water Requirement Satisfaction Indices (WRSI) developed by FAO and FUSENET. We suggest you directly refer to these sources (http://www.fao.org/docrep/X0490E/X0490E00.htm) for information about these types of models and example crop parameters. These models perform an accounting exercise, adding up the amount of rainfall stored by the soil and subtracting out the water that is removed by the plant through evaporation and transpiration as grows over the season. If the amount of water available in the soil falls below what the plant is able to withdraw, the model predicts reduced yields. The model allows you to mimic the enhanced impacts of water stress during times when the crop is particularly sensitive to water stress using additional parameters that are often not included in traditional WRSI calculations.

### Prerequisites

In order to run this module, you must have already entered a precipitation dataset as well as a potential evapotranspiration dataset. See the "Managing Data" section for information on how to upload these datasets.

### Running Crop Water Stress Accounting

To begin the Crop Water Stress Accounting module, select a Precipitation dataset and a Potential Evapotranspiration dataset from the brown area on the left side of the page.
The next step is to enter parameters describing the growing season and other crop and soil specific parameters. You may select a Parameter dataset from the pulldown menu to choose an initial default set of parameters that can be edited as you choose, or you may select “New” to create a new parameter dataset from scratch.

Parameters describing the growing season include:

- **Potential Sowing Window**: This is the estimated period in which conditions are likely to be appropriate for a farmer to sow the crop being taken into consideration. This can be determined from agrometeorological information as well as feedback from farmers. It is a measure of the atmosphere’s ability to remove water from the surface through the processes of evaporation and transpiration, assuming there is no control on water supply.

- **Sowing Requirement**: The Sowing Requirement is the amount of cumulative rainfall (in mm) received during the sowing window that is necessary to establish adequate conditions for a farmer to sow.

The Crop and soil Specific Parameters include:

- **Water Holding Capacity (WHC)**: The WHC refers to the water holding capacity of the soil in the cultivated area surrounding the station that falls in the same micro-climate as the station.

- **Crop Coefficient (Kc)**: The Kc values define the changing ability of the crop to remove water from the soil through evaporation and transpiration. Often, it reflects increases in the leaf area of the crop as it grows.

- **Varying Yield Response Factors (Ky)**: The Ky value allows you to model the enhanced impacts of water stress during the times when the crop is particularly sensitive to water stress. This is typically not included in traditional WRSI models. This additional parameter is important because many crops, such as Maize, may have periods in which they have a large leaf area and are not vulnerable to drought and other periods in which they have the same amount of leaves but are very vulnerable to drought.

- **Seasonal Ky (SKy)**: The SKy describes the reduction of relative yield according to the reduction in the crop’s evapotranspiration caused by soil water shortage over the course of the growing season. The SKy is specific to each crop.

After choosing the precipitation and potential evapotranspiration datasets and selecting the agrometeorological parameters, you are ready to run the simulation.

The software requires complete evapotranspiration data to operate, but can have missing data in the precipitation datasets. For transparency, a conservative strategy is used for missing data. Any years for which there is missing precipitation data in the sowing window or growing season will be removed from the analysis. If you would like to use another approach to handle missing data, you can fill the missing data using your approach and upload that dataset for analysis.

It is important to remember that many crops will not grow in many climates. Therefore, you will need to enter parameters and sowing conditions that are realistic for the climate represented by your rainfall and evapotranspiration data. Otherwise, the module may determine that the crop has failed, returning few, if any years of simulated crop water stress.
Results

The module results appear in the form of a graph displaying the Crop Water Index, which is the crop productivity estimate. A table in the results page also displays the estimated sowing dekad for each year, as well as the sowing condition in millimeters. If the tables and charts are empty or have very few entries, it is likely that there was not sufficient rainfall for successful sowing in most of the years.

The results of the Crop Water Stress Accounting Module can be saved for use in the Loss Conversion Module, which uses the Crop Water Index to estimate losses.

Loss Conversion Module

The Loss Conversion Module allows you to take the output of the Crop Water Satisfaction Module and convert that into a simple approximation of financial losses due to low yields. For contract design, this conversion does not need to be extremely accurate, but simply to reflect that the years with low yields have large losses.

First, select a Crop Water Index dataset. This could be a dataset produced in the Crop Water Accounting Module or it could have been an actual yield dataset uploaded through the Manage Data module via the Productivity Dataset upload.

After selecting a productivity dataset, enter the scaling and threshold percentile parameters. Alternatively, select a scaling and threshold percentile using the Saved Parameters button.

Scaling

The scaling parameter is provided to allow you to enter a multiplier to scale loss calculations. If you want to roughly reflect the cash value of losses, you might think of the crop water stress estimates as a fraction of a typical ‘good’ yield. For example, if a typical ‘good’ yield is 100KG, and the approximate value of the crop is 10 schillings per kg, you might set the multiplier to 1000.

At best, this will be a rough proxy for losses, and unless it is off by orders of magnitude, it is unlikely to impact the contract evaluation and design noticeably. You can experiment with alternate multipliers if changing the multiplier has any impact on the contract you select. The default scaling parameter is set to “1000”.

Threshold Percentile (a number between 0 and 1)

Because insurance typically protects against substantial losses, there is a parameter that allows you to focus the analysis and contract evaluation results on the years that are relatively bad. The designer can select how bad a year must be to be categorized as a loss year. The threshold parameter allows the designer to select which fraction of the years is categorized as losses. The threshold parameter is a number between 0 and 1 that is used to make this selection. Any years that are in the fraction above the loss threshold, are set to zero loss for the analysis. For example, if the threshold parameter is set to 1, all years are included in the analysis. If the threshold is set to 0.5, the calculated losses for the half of the years that have the highest losses are included in the analysis and all other years are set to a loss of zero. In other modules you will try and match up the years in which the farmers experienced losses with the years that the contract experiences payouts.
Results

The results page of the Loss Conversion Module displays the harvest year and losses for each harvest year. If these years are not the years for which you know that farmers experienced losses, you may return to the Crop Water Stress Accounting module and modify the crop parameters or growing season.

If you are satisfied with the results, you may save them for later comparison with the contract you will design in the Create Contract module.

Create Contract Module

In the Create Contract Module, you can create a contract and apply it to a historical rainfall dataset to see what payouts would have occurred based on the rainfall.

The Create Contract Module applies an index to a historical precipitation dataset to determine when payouts would have occurred had that contract been in place. It allows you to create a contract and also calculates what payouts would have occurred using that contract on a precipitation dataset. In addition, you can use the Create Contract Module to calculate payouts for contracts that you have created previously.

For the Create Contract Module, you can choose what currency units you wish to work with. However, you should be consistent with the choice you make across all of the modules.

For example, you could choose to set the liabilities to the actual cash value in the local currency. Alternately, because the cash value of the sum insured may not be known during the initial contract design, you can select simple units that can be scaled to any maximum liability in any currency. For example, the defaults in the module are set to 1000 to allow for calculations that can be scaled to any currency and any maximum liability.

As an illustration, if the liability is later determined to be, say, 4402 Kwacha, then the Webtool payouts and liabilities provide by the module can be scaled by multiplying by (Actual Maximum Liability /1000). Thus, a payout in the module output of 500 would be 500 * 4402/1000 = 2205 Kwacha. You should choose the strategy that is most convenient for you.

The module is designed to allow contracts with multiple phases, each with a different length and rainfall trigger and exit. For each phase, the annual payout will be calculated using the following formula:

\[ Payout = (1 - \frac{(Rainfall \text{ Sum} - Exit)}{(Trigger - Exit)}) \times Max \text{ Payout} \]

You will be allowed to customize the contract further in the Pricing module.

1. To begin, select a precipitation dataset.

2. Next, you must choose a set of contract parameters. You may select a previously saved set of parameters from the Saved Parameters menu, or enter new ones.

Contract parameters include the following:

Contract Start Window

In the contract parameters, the contract window is a time period consisting of several dekads, each of which is the potential start time for the contract. Once the contract start requirement is met within this time period, the first phase of the insurance contract begins. If the requirement is not met during this time period, it is assumed that there was insufficient rainfall to initiate the rest of the contract and for that year the contract pays the value you entered into the contract failed start liability box.
**Contract Start Requirement**
In the contract parameters, the contract start requirement is the amount of cumulative rainfall (in mm) received during the contract start window that is necessary to establish adequate conditions for a farmer to sow. If the contract start requirement is not met during any dekad of the contract start window, a payout for that year will be the value you enter in the failed start liability box below.

**Phase Length and Timing**
Contract calendars are divided into several phases, in some contract design strategies, each phase corresponds to a particular stage of the crop’s growth. In other strategies, phases may simply reflect a rainfall total, for example to provide a proxy for the approximate strength of the beginning of the rainfall season. In either case, a phase lasts for the number of dekad selected by the designer. Each phase has its own payout function consisting of a Trigger, Exit, and Maximum Payout.

**Trigger**
The Trigger is an amount of rainfall (in mm) below which a farmer will receive a payout. It is the maximum amount of rainfall that can be received and still constitute a payout for the farmer. Each phase of the contract has its own individual trigger which may be configured to your discretion. If you are having trouble, the Tune Contract module will help configure your triggers for the desired premium.

**Exit**
The exit is the point in a phase’s payout function at and below which the farmer receives the maximum possible payout for that phase. It is an amount of capped rainfall (in mm) occurring during the given phase. This parameter is not determined scientifically to represent a point of complete loss, but can be used to target insurance in the most cost effective manner.

**Maximum Liability**
Maximum Liability is the total amount of money insured. If the rainfall is below the exit for that phase, the Maximum Liability will be the payout for that year. Otherwise, the payout will be a percentage of the Maximum Liability. If the total payouts from multiple phases exceed the contract Maximum Liability, the total payout is capped at the contract Maximum Liability.

**Dekadal Cap**
The Dekadal Cap is the limit placed on dekadal rainfall totals. It is included in case there is a short period with a great deal of rainfall during an otherwise very dry period. Using the dekadal cap, a contract designer can set the contract so that excessive rainfall during any single dekad can be disregarded so that it does not inappropriately impact the phase total. Any amount of rainfall received above the Dekadal Cap in a ten-day period (dekad) is disregarded in payment formulas.

3. After selecting a precipitation dataset and contract parameters you are ready to calculate payouts. When you are ready to continue, click on the “Run Simulation” button on the bottom right of the screen.

**Results**

The results page of the Create Contract Module displays a graphic showing the payouts for the dataset, as well as a table displaying the sowing dekad, rainfall for each phase of the contract, payout for each phase of the contract, and total payout.
For transparency, a conservative strategy is used for missing data. Any years for which there is missing precipitation data in the sowing window or phases will be removed from the analysis. If you would like to use another approach to handle missing data, you can fill the missing data using your approach and upload that dataset for analysis.

As with the water stress calculation module, it is possible to have contracts that are not suitable for the climate represented by your precipitation dataset. In this case, you may have full payouts in every year or no payouts at all. To solve these problems, it is important to take the typical precipitation conditions into account when designing a contract.

**Pricing Module**

The Pricing Module will utilize the payout calculations from the Create Contract module to calculate a ‘pseudo’ price of the contract. This ‘pseudo’ price uses a simple pricing formula based on calculated payouts, a user selected loading parameter and value at risk. The end result will be price calculations and a complete package of contract information available for further analysis in the Tune Contract and Contract Evaluation modules.

The intent of the ‘pseudo’ price is to allow the designer to model risk protection and insurance cost tradeoffs sufficiently to make quality design decisions. The designer should be aware that it is a working price for design purposes and is likely to be somewhat different than the final price of a transacted contract.

The actual price of the insurance will most likely be a price negotiated between the project stakeholders, which may be calculated using different formulas than are built into the pricing module. Often, these are driven by proprietary analysis done by the insurance companies. In addition, the price often includes administrative and logistical costs.

Our calculation of the “pseudo price” is as follows:

\[
\text{Average payout} + Loading \times (\text{Value at Risk} - \text{average payout}).
\]

For the pricing module, you can choose what units you wish to work with. However, you should be consistent with the choice you make across all modules. For example, you could choose to set the maximum liabilities to the actual cash value in the local currency. Alternately, because the cash value of the sum insured may not be known during the initial contract design, you can select simple units that can be scaled to any maximum liability in any currency.

**Example 3.1. For Example**

The defaults in the module are set to 1000 to allow for calculations that can be scaled to any currency and any liability. As an illustration, if the maximum liability is later determined to be, say, 4402 Kwacha, then the prices provided by the module can be scaled by multiplying by (Actual Maximum Liability /1000).

Thus, a cash price in the module output of 500 would be 500 * 4402/1000 = 2205 Kwacha.

You should choose the strategy that is most convenient for you.

The first step is selecting a Payout dataset; these are the results you saved in the Create Contract module.

Select a pre-saved set of parameters from the "Saved Parameters" menu, or enter your own. The pricing parameters are:

- **Maximum Liability**: Maximum Liability is the total amount insured. In some unusual circumstances, this may be different from the Maximum Liability selected for the contract creation modules. To allow for these situations, the pricing module requires that the designer enter the
Maximum Liability into the pricing module. Otherwise, use the same Maximum Liability you used for Create Contract.

Value At Risk (v.a.r.) (a number between 0 and 1): This is the percentage value of payout for which the insurance company must hold assets and is typically set to 0.99, to reflect the size of a one in one hundred year loss.

For example, entering 0.99 for Value at Risk will indicate the 99th percentile is the relevant risk. Entering 0.99 will lead the software to calculate the size of the payout that happens in 1 out of 100 years. Since the insurance company will need to hold assets to provide payouts for this event, the loading (described below) is applied to this amount (less the average payout) to estimate the financial cost of securing finances to honor the Value at Risk sized payout.

Loading: This is the rate applied to the value at risk (less average payout) to calculate the financing costs to the insurance company for the contract. This reflects the costs that the insurance company must pay to obtain sufficient financing to be able to make large payouts. It is commonly set to between 0.05 and 0.15.

Results

The results of the Pricing Module are displayed on the Results page, in table form. The following results are presented in the table:

Premium in Cash: The actual amount of money that will be charged for active insurance coverage based on the pseudo price formula.

Premium as a percentage of sum insured: The premium as a fraction of the maximum liability.

Value of Loss at Value at Risk Percentile: The size of the payout at the value at risk. For example, if the value at risk is 0.99, this is the computer’s estimation of the cash payout size for the 1 in 100 year payout. If there are less than 100 years of data, this is likely to be underestimated. The rainfall simulation module can be used to help address this type of problem.

Payout Variance: The statistical value of the variance in payouts from year to year. A larger value indicates a widely varied contract, whereas a smaller value indicates a contract for which the payout does not vary greatly from year to year.

Saving the results saves the Pricing parameters, which are not used in any other modules, but will be useful when running the module again at a later time.

Tune Contract

The Tune Contract module will optimize the triggers in each phase of the contract to a “pseudo price” entered by the user. This optimization will minimize the variance in losses from the Loss Conversion module and the insurance payments from the Pricing module subject to the “pseudo price” constraint. This unofficial “pseudo price” is expressed as a percentage of maximum liability and follows standard, transparent risk pricing methods to provide the optimization software to approximate contract coverage and contract price trade offs for design purposes. Of course, the final price of the contract will ultimately
be determined in negotiations between stakeholders, and is likely to include additional components, such as administrative and logistical costs. The pseudo price selected for design should be set low enough so that the contract is workable with these additional anticipated costs.

Our calculation of the “pseudo price” is as follows:

\[ \text{Average payout} + \text{Loading} \times (\text{Value at Risk} - \text{average payout}). \]

The Tune Contract module will determine the relative levels of the triggers for each phase. In other words, the optimizer minimizes the variance of losses that a hypothetical farmer would face if that farmer had purchased the insurance by adjusting the triggers while maintaining a maximum insurance pseudo price. In order to ensure the price must not go above the constraint, when the optimizer raises the level of one trigger, it lowers the levels of the others.

It is important to think of the tuner as a tool as opposed to something that will magically provide you with the perfect contract. Many real world objectives, goals, and constraints are not modeled by the tuner, so you must keep the real world set of performance issues in mind when you decide how to use the tuner’s results.

The tuner performs a classic numerical optimization. It is important to keep in mind that this exercise is a challenging computational task, and it is not guaranteed that the module will successfully find an improved contract.

The strategy that the module uses is to make very small changes in the contracts, adjust the contract so that it meets the price constraint, and then see if the new contract performs better than the original. If improvements are seen, the computer continues to change the contract in the same direction until no more improvements are evident. If the computer is unable to see improvements with small changes, or is unable to adjust the contract to meet the price constraint, the tuning exercise will fail.

When the tuner fails, there are several strategies you can use. One strategy is to increase the price constraint for the contract until the tuner works, and then use the results from the tuner in new contracts as you gradually reduce the price. Changing your contract to have much higher triggers may put the tuner in a situation for which it can understand how to navigate to improve things. You can also use the Rainfall Simulator to generate many more years of rainfall data, and to repeat the Crop Productivity and Contract modules. This generates datasets that, although slower to compute with, may be easier for the computer to detect improvements by trying small changes.

In addition, the tuner will stop once it stops seeing improvements when making small changes. Because of this, it will typically find the best contract that is ‘in the neighborhood’ of the original contract. Therefore, it will be valuable to try the tuner with radically different contracts and then identify the best of the best.

**Note**

As of this writing (December 2009) the Tune Contract module is only designed to function for contracts that have more than one phase.

First, select a Payout dataset, a Precipitation dataset and a Loss dataset.

For the Payout dataset, use the results that were saved in the Pricing module.

In the Loss dataset selection menu, use the results saved from the Loss Conversion module. The Tuning Module will discard any years for which either Payout or Loss data does not exist.

Next, select a set of previously saved Tune Parameters from the pull down menu or create your own. Tuning parameters are:

- **Target Price** The target price is the desired annual premium.
Value at Risk  
This is the percentage value of payout for which the insurance company must hold assets. For example, entering 0.99 for Value at Risk will indicate the 99th percentile is the relevant risk. Entering 0.99 will lead the software to calculate the size of the payout that happens in 1 out of 100 years. Since the insurance company will need to hold assets to provide payouts for this event, the loading (described below) is applied to this amount (less the average payout) to estimate the financial cost of securing finances to honor the Value at Risk sized payout.

Loading  
This is the rate applied to the value at risk (less average payout) to calculate the financing costs to the insurance company for the contract.

Illustrative Example

Example 3.2. Example

For illustration, our analysis used a Value at Risk that is the 99th percentile and a loading of 6%. Unless there are several hundreds of years of data, the computer cannot be relied upon to accurately estimate the size of the 99th percentile (the 100 year event). However, it still represents a measure of the risk in the distribution and functions as a simple and transparent index that that may be sufficient for the optimizer to select appropriately between contract possibilities. The rainfall simulator can be used to improve estimates by simulating hundreds of years of rainfall. Alternately, a contract can be designed so that the value of risk is at a much more frequent probability.

The loading of 6% for the Value at Risk typically differs from final pricing and is chosen by the insurer given his portfolio and required return on VaR. A value of 6% is chosen simply to indicate some element of risk margin. Price differences should only have a subtle effect, if any, on the design of the contracts so long as the final insurance pricing is qualitatively similar. Of course, after the contract is designed, for accurate pricing of the final product, it is important to have more complete processes for determining the final price, determined through discussions on costs and pricing from insurers and other stakeholders.

Results

The results of the Tune Contract Module are displayed on the Results page, in table form. This includes the new triggers and exits produced for each phase by the Tune Contract module along with the rest of the contract parameters used to obtain the results.

Saving these results preserves this new set of contract parameters, which are identical to the results of the Create Contract module and may be used in any module that those results are used, such as the Pricing module. In this manner, you may compare both contracts’ performance using tools such as the Contract Evaluation module.

Contract Evaluation

The Contract Evaluation Module provides an assessment of the contract’s performance by comparing a payout dataset from the Create Contract module and crop loss dataset from the Crop Water Stress Accounting and Loss Conversion modules. Because there is no single parameter that completely represents the quality of the performance of a contract, the module provides a set of useful statistics. For example, in some cases it may be important for the contract to closely track losses when there is a payout. In other cases it may be more important to have the payouts occurring in the correct years. Most likely, a balance of these kinds of things will be important. This module will provide you with you with important indicators as you discussions with clients to determine what kinds of performance the contracts should have, and how well performs on those fronts.
To begin, select both a Payout and Loss Conversion dataset. Datasets listed in the Payout Data selection menu are the results of previous runs of the Pricing module. Likewise, those datasets in the Loss Conversion selection menu are the results of previous runs of the Loss Conversion module. The Module will discard any years for which either Payout or Loss data does not exist.

There is no additional parameter selection for the Contract Evaluation Module, so you are ready to evaluate your contract.

**Results**

**Correlation**

The correlation between the losses and the payouts. This is a number between -1 and 1, which will indicate how closely related the contract matches with crop losses. The higher the number is, the more closely related the years with insurance payouts are with the crop losses calculated in the Loss Conversion module. Although there is no objective level that the correlation must attain, if this number is too low, it is a signal that you may need to adjust your contract.

**Number of Payouts**

The number of payouts to occur during the total number of years to which the contract was applied.

**Pay Rate**

The frequency of payouts.

**Mean Payout**

The average payout, including years in which there was no payout.

**Mean of Payouts greater than zero**

The average payout for years in which a payout occurred.

**Percentage Drop in Variance of Losses-Payouts**

The decrease in variance of losses-payouts. This will indicate how well your contract is helping farmers smooth income. If the contract performs well at reducing farmers’ losses, the drop in variance will be strongly negative. This will be very sensitive to specific assumptions in determining losses and payouts, so it is best used to make comparisons between different versions for the same contract, holding loss data and precipitation data the same.

**Percent of Payouts in Worst 1/2 (or 1/3 or 1/4)**

This is an indication of how well the insurance is targeting the correct years for payouts. It is the percentage of payouts that are made in the one-half (or one-third, or one-fourth) of the total number of years with the greatest losses.

You may save these results for later reference; however they are not used in other modules.

**Rainfall Simulator**

The Rainfall Simulator will simulate hundreds of years of rainfall data based on the statistical properties of the precipitation data set that is selected. This module is useful in places where there is very little historical rainfall data and it uses methods that automatically build in additional variation in the rainfall to reflect increased uncertainty due to limited length of datasets.

When you use the historical data to design and evaluate your contract, this is known as ‘historical burn’ analysis. It is extremely transparent and can be easily communicated to stakeholders. For this analysis, the probability distribution of the indexed parameter is implied entirely by past measurements. Although useful, when applied without other analyses this approach has limitations. For example, one or two major...
events can distort the probabilities, while any event that has not happened in the historical record is not considered.

Because of the limitations of historical burn analysis, it is typically complimented with rainfall modeling and simulation. Using rainfall data that are available, plus an understanding of the variables that influence rainfall, the modeling generates hundreds of years of synthetic rainfall data, which include events that are possible but have not actually occurred in the past. This approach can be used to extend limited datasets, allowing and more accurate estimates of diagnostics for contract performance, less idiosyncratic contracts, and often improved performance of the Tune Contract Module.

With the simulated rainfall from this module, you can re-run the other modules to get an improved understanding of losses, payouts, and prices to arrive at more robust contracts.

The module has the following qualities:

1. It does not take a long time to fit the model to data

2. It is parametric, so that it can simulate values that did not occur in the observed data, which is impossible using certain resampling strategies

3. It is sensitive to the amount of data with which it is fit, so that less observed data results in more uncertainty in simulated rainfall. To do this conveniently, we use the Bayesian framework

4. Lastly, the procedure is automatic, so that as little supervision and data setup as possible is required. This might be the greatest challenge of all, because different amounts of observed data (5 years vs. 50 years, for example) and different rainfall generating processes throughout the world may require different types of models

The data sets that the module generates are available to be used in any other module, as they are considered precipitation data sets. The only difference is that instead of having years numbered historically, the years will be numbered from 1 to several hundred. Also, needless to say, these data sets can slow the processing of some of the modules.

To reach acceptable performance and reliability, the simulator does not model daily rainfall. Instead, it models rainfall over 10 day periods. The outputs of the simulator therefore have identical rainfall for each day within any given dekad. Although this simplification does not impact contract payouts or pricing, it is likely to impact the water stress calculations. Because it does not represent variation within a ten day period, the water stress module may underestimate the amount of stress that a crop may face due to daily variation. In cases where this simply leads to a systematic underestimate of water stress, this will not substantially impact contract design. However, the designer should keep this shortcoming in mind.

There are many other situations in which a rainfall simulator does not accurately reflect true rainfall statistics, so you should always view results from any rainfall simulator with a critical eye. To assist you with this task, the simulator generates a set of reports that help you assess its performance.

For information on how to interpret these reports, you can refer to (Rainfall Simulator Technical AppendixA.pdf [AppendixA.pdf])
Chapter 4. Managing Data

This module is designed to allow authorized users to upload and download data, as well as perform account management and classroom administration. Authorized users are Teachers and Administrators.

Manage Data

From this page you can upload and download the set of data in WIIET’s database that are marked “downloadable” and “sharable,” as well as any “not sharable” datasets that you uploaded, to your personal computer. Each of the modules provides additional files for downloading. Depending on your downloading needs, you may chose to download some information directly from the Modules themselves. Types of datasets available for download from the Manage Data Module include.

Download or Delete Data

Selecting the Download button for any of the datasets should save the Comma-Separated-Values (csv) file to your desktop. You should be able to open this with any spreadsheet-capable application, such as Microsoft Excel.

The Delete button will remove the dataset from the database, making it completely unavailable to any module in the future.

Uploading Data

Teachers and Administrators can upload datasets for use in WIIET’s different modules. You may mark a dataset you upload “Downloadable for Students” if you wish to share them with students of your class. There are four types of data that can be uploaded, each of which are described in the following sections.

<table>
<thead>
<tr>
<th>Name</th>
<th>Enter the name of the dataset that will uniquely identify it, so that others can understand exactly what it represents</th>
</tr>
</thead>
<tbody>
<tr>
<td>Description</td>
<td>Enter the description of the dataset that will help identify it.</td>
</tr>
<tr>
<td>File Name</td>
<td>Use the 'Browse' button to find a dataset on your computer to upload.</td>
</tr>
</tbody>
</table>

The data should be in a Comma-Separated-Value (CSV) format. You can create this type of file by exporting it from Excel.

File Formats

All datasets used by WIIET are in CSV format, which are comma-delimited text files. These files may be edited by Microsoft Excel or any similar spreadsheet software, as well as in any text editor.

When opened in Excel, CSV files will appear as identical to proprietary Excel files, but care must be taken to save the files in proper CSV format. You can do this in the “File/Save As” menu, by selecting “CSV (comma-delimited) (*.csv)” as the file type.
Example 4.1. Excel Example

Table 4.1. Precipitation Data (Excel Table)

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>22-Feb</td>
<td>0</td>
<td>0</td>
<td>5</td>
<td>0</td>
</tr>
<tr>
<td>23-Feb</td>
<td>5</td>
<td>0</td>
<td>5</td>
<td>20</td>
</tr>
<tr>
<td>24-Feb</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>25-Feb</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>26-Feb</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>27-Feb</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>27-Feb</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>28-Feb</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>29-Feb</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>01-Mar</td>
<td>0</td>
<td>2</td>
<td>0</td>
<td>8</td>
</tr>
<tr>
<td>02-Mar</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>03-Mar</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

In a text editor, the CSV will appear in raw format, with each data element separated by a comma. For example, the payout data series formatted in the two left columns would appear in raw CSV format as appears in the rightmost column.

Table 4.2. Precipitation Data: File Format

When it is possible, the easiest way to get data into the correct format is to download a dataset from the WIIET and replace the data in it with the information you would like to use, and then uploading the new file.

**Precipitation Data**

These are daily rainfall data series. The data must be formatted with the day of the year as the row labels (this equals 366 days, the leap day of February 29th is included and left blank in those years it does not occur) and the year as the column labels. This file must be a comma-separated-values (CSV) file!

**Potential Evapotranspiration Datasets**

Evapotranspiration datasets are dekadal and regional. The data must be formatted with the dekad in the column heading and the station name in the leftmost row.

<table>
<thead>
<tr>
<th>Station</th>
<th>1-Jan</th>
<th>2-Jan</th>
<th>3-Jan</th>
<th>1-Feb</th>
<th>2-Feb</th>
<th>3-Feb</th>
<th>1-Mar</th>
</tr>
</thead>
<tbody>
<tr>
<td>Lilongwe</td>
<td>45</td>
<td>43</td>
<td>42</td>
<td>40</td>
<td>38</td>
<td>39</td>
<td>42</td>
</tr>
</tbody>
</table>

**Productivity Datasets (Crop Water Index)**

These are crop productivity data sets, including historical and simulated yield data. The data must be formatted in two columns: the first is the Harvest Year and the second is the Yield Index or productivity data.

<table>
<thead>
<tr>
<th>Harvest Year</th>
<th>Yield Index</th>
</tr>
</thead>
<tbody>
<tr>
<td>1962</td>
<td>0.808082</td>
</tr>
<tr>
<td>1963</td>
<td>0.731678</td>
</tr>
<tr>
<td>1964</td>
<td>0.842899</td>
</tr>
<tr>
<td>1965</td>
<td>0.87507</td>
</tr>
<tr>
<td>1966</td>
<td>0.62957</td>
</tr>
<tr>
<td>1967</td>
<td>0.559885</td>
</tr>
<tr>
<td>1968</td>
<td>0.730421</td>
</tr>
<tr>
<td>1969</td>
<td>0.849413</td>
</tr>
<tr>
<td>1970</td>
<td>0.71208</td>
</tr>
<tr>
<td>1971</td>
<td>0.861556</td>
</tr>
</tbody>
</table>
Payout Dataset

The data must be formatted in two columns: the first is the Harvest Year and the second is the payout in that year.

<table>
<thead>
<tr>
<th>Harvest Year</th>
<th>Payout</th>
</tr>
</thead>
<tbody>
<tr>
<td>1961</td>
<td>1000</td>
</tr>
<tr>
<td>1962</td>
<td>1000</td>
</tr>
<tr>
<td>1963</td>
<td>1000</td>
</tr>
<tr>
<td>1964</td>
<td>1000</td>
</tr>
<tr>
<td>1965</td>
<td>1000</td>
</tr>
<tr>
<td>1966</td>
<td>1000</td>
</tr>
<tr>
<td>1967</td>
<td>1000</td>
</tr>
<tr>
<td>1968</td>
<td>1000</td>
</tr>
<tr>
<td>1969</td>
<td>1000</td>
</tr>
<tr>
<td>1970</td>
<td>1000</td>
</tr>
</tbody>
</table>

Download Data from Data Library

This is a link to IRI’s Data Library. In the Data Library you may find Precipitation, Evapotranspiration, and Productivity data that you wish to download in order to upload for use in WIIET. IRI’s Data Library also includes many other valuable datasets that may be helpful in the contract design process.

Classrooms

Create a Classroom

This is where you organize your learning environment. Creating a classroom allows you to select what data you are going to make available for your own use during your sessions working with WIIET, as well as what data you are going to make available for the students you are instructing.

When creating a classroom, you must select Precipitation and Potential Evapotranspiration Datasets that you want to make available during the Tool’s use. You may also select Productivity Datasets. The menu for each type of data includes any data you have uploaded to the tool, as well as any data uploaded by other users with the role of Teacher or Administrator that was marked “sharable.” You may select as many datasets as you wish from each of the data selection menus to include in your classroom.

After selecting the datasets that you want to be available to you and your students during you sessions using WIIET, you are ready to create a classroom. Clicking on the “Create a Classroom” button at the bottom of the screen will bring you to the “Classroom Confirmation Page.” This page tells you your class identification information, the usernames for your students and their passwords, and the datasets available to the class for use in WIIET.

View/Modify a Classroom

From this screen, you can view or modify the classroom you created. When you click on the “Create a Classroom” link, you will be given a Class ID that WIIET uses to keep track of your data. Use this number
to reference the classroom you created. It is possible to change the teacher assigned to the classroom as well as the datasets available to that class, or to delete it entirely.